Magyar Számítógépes Nyelvészeti Konferencia

MSZNY 2003

Szeged, 2003 december 10-11
http://www.inf.u-szeged.hu/mszny2003

Szegedi Tudományegyetem Informatikai Tanszékcsoport
6720 Szeged, Árpád tér 2.

Magyar Számítógépes Nyelvészeti Konferencia

MSZNY 2003

Szeged, 2003 december 10-11
http://www.inf.u-szeged.hu/mszny2003

Szegedi Tudományegyetem Informatikai Tanszékcsoport
6720 Szeged, Árpád tér 2.

Kiadó: Szegedi Tudományegyetem, Egyetemi Nyomda
Szerkesztette: Dr. Alexin Zoltán és Csendes Dóra
Szeged, 2003. november

Előszó

A Szegedi Tudományegyetem Informatikai Tanszékcsoportja 2003. december 10-11 között rendezi meg az első Magyar Számítógépes Nyelvészeti Konferenciát (MSZNY 2003) Szegeden. A konferencia fő célja, hogy a számítógépes nyelvészet és beszédfeldolgozás területén végzett kutatásoknak és azok eredményeinek ismertetésére fórumot biztosítson. A tervek szerint a konferenciára (a régi Neumann kollokviumi hagyományokhoz hasonlóan) éves rendszerességgel kerülne sor.
Az idei esemény a területen végzett munka eddigi és legaktuálisabb eredményeit mutatja be, a cikkek és előadások elsősorban számítógépes módszerekkel megoldható nyelvészeti problémákkal foglalkoznak.
A felhívásokra érkezett cikk kivonatok közül a Programbizottság 41-et fogadott el teljes cikk beküldésére és hosszú előadás megtartására, és további 20-at rövid előadás megtartására. A konferencia jelen kiadványában 39 teljes cikk és azok rövid angol nyelvű kivonata, valamint 20 rövid előadás magyar és angol nyelvű kivonata jelenik meg.

Ezúton szeretnénk köszönetet mondani a konferencia Programbizottságának: Vámos Tibor programbizottsági elnöknek, valamint Gordos Géza, Prószéky Gábor és Váradi Tamás programbizottsági tagoknak. Szeretnénk továbbá megköszönni a Rendezőbizottság: Alexin Zoltán, Csendes Dóra és Gyimóthy Tibor munkáját.

Csirik János
a Rendezőbizottság elnöke
2003. november

Tartalomjegyzék

Hosszú előadások

Rövid előadások

Korpusz-egyértelműsítés – a morfoszintaxison túl

Nagy Viktor

MTA Nyelvtudományi Intézet, Budapest
nagyv@nytud.hu

Kivonat A korpuszok morfoszintaktikai egyértelműsítése nem elegendő ahhoz, hogy a korpusz minden szövegszaváról megállapítsuk, mely lexémához tartozik, a probléma megoldásához szemantikai egyértelműsítés is szükséges. A dolgozat bemutatja, hogy a más nyelvekre már sikeresen alkalmazott statisztikai algoritmusok biztató eredményeket produkáltak már kis tanulómintán is.
Kulcsszavak: statisztikai alapú szójelentés-egyértelműsítés, homonímia-egyértelműsítés, felügyelt tanulás, naiv Bayes-osztályozás, döntési lista algoritmus

1. Bevezető

A napjainkban használatos korpuszoknak egyszerre kell teljesíteniük azt a két követelményt, hogy nyelvileg interpretáltak, illetve megfelelő méretűek legyenek. Bizonyos méret felett már a minimális annotáció sem végezhető el emberi munkával. A 150 millió szavas Magyar Nemzeti Szövegtár (MNSZ) [1] morfoszintaktikai elemzése és egyértelműsítése (tagging) – tanítás után – teljesen automatikusan zajlott. A morfoszintaktikai egyértelműsítés a korpusz szövegszavaihoz hozzárendeli a szóalak alapalakját (lemmáját), szófaját és morfoszintaktikai kategóriáját. Ez az eljárás a szövegszó morfoszintaktikai környezetét veszi figyelembe, vagyis a környezetet a konkrét szövegszóktól elvonatkoztatva morfoszintaktikai kódsorozatként látja, ezért nem tud olyan jelenségeket megkülönböztetni, amelyeknek szemantikája különbözik, morfoszintaktikai disztribúciója viszont azonos [2].

Az ilyen mélységű elemzés általában nem elegendő annak meghatározására, hogy a szövegszó mely lexémához tartozik, mivel a (lemma, szófaj) párhoz nem mindig rendelhetünk egyértelműen egyetlen lexémát, vagyis a magyar lexikográfiai gyakorlat által önállónak tekintett szótári egységet. Ez a szófajon belüli homonímia esetén figyelhető meg: míg a morfoszintaktikai egyértelműsítés el tudja dönteni, hogy a *tűz* alak a *tűz*[1] IGE vagy a *tűz*[2] FN előfordulása-e egy adott kontextusban, az *ül* (1 nyugszik', 2 'ünnepel') vagy a barát (1 'társ', 2 'szerzetes') homonimáit nem tudja szétválasztani, mivel az utóbbiak morfoszintaktikai környezete megegyezik.[1]

További problémát jelent az az eset, amikor egy szóalak több különböző, de azonos szintaktikai viselkedésű lemmához tartozhat. A *sejtette* lehet a *sejtet* IGE

[1] A homonimitás eldöntéséhez az ÉKsz-t vettük alapul, és a homonimaszámozás is az ÉKsz számozását követi.

vagy a *sejt* IGE tárgyas múlt idejű egyes szám 3. sz. alakja, a *lappal* pedig a *lap* FN vagy a *lapp* MN/FN eszközhatározós alakja. Ilyenkor az egyértelműsítő nem tud a szóalakhoz egyetlen lemmát rendelni. A többértelmű lemmák egy része mégis mutat valamelyest kontextusbeli különbséget. A *bán* IGE és a *bánik* IGE különböző vonzatkerettel rendelkezik, tehát elviekben meg lehet őket különböztetni csupán a morfoszintaktikai környezet alapján. Az MNSZ-ben használt taggelési eljárás azonban Markov-modellre épül, ezért szükségszerűen kicsi a figyelembe vett környezet mérete, ami lehetetlenné teszi a mondatban szétszórt vonzatok figyelembe vételét.

Ahhoz, hogy a morfoszintaktikai egyértelműsítés fent említett korlátain túllépjünk, mindenképpen az elvont morfoszintaktikai kategóriák mögé kell nézni a környezetben, és mivel az eldöntendő problémák egy része szemantikai, a szemantikai egyértelműsítés problémáját kell megoldani.

2. Statisztikai alapú szójelentés-egyértelműsítés felügyelt tanulás alapján

A szójelentés-egyértelműsítés (word sense disambiguation, WSD) célja annak megállapítása, hogy egy szó melyik lehetséges jelentésével fordul elő egy adott kontextusban. Az egyértelműsítésnek azt a fajtáját vesszük most figyelembe, amelynél a feladat szóelőfordulások előre felállított jelentésosztályokba sorolását jelenti. Fel kell tételeznünk, hogy a szavak jelentésosztályainak száma véges, és egymástól jól elkülöníthetők, de hogy pontosan mennyi is van, azaz milyen aprólékos a jelentésfelbontás, az alkalmazásról alkalmazásra változik. Az eredeti feladatunk szerint a jelentésosztályok lexémák, vagyis – a probléma típusától függően – a szó lehetséges lemmái vagy lehetséges homonimái vagy ezek kombinációi. Ez utóbbira példa az *érnek* igei alak, amely a *ér*[1] 'érint', *ér*[2] 'értéket képvisel, érvényes' és *érik* lexémát is képviselheti. Mivel a morfoszintaktikai egyértelműsítő által adott kimenetből indulunk ki, az *ér*[3] főnévi lexéma már nem jöhet szóba.

Az eljárás, amelyet a fent vázolt egyértelműsítési problémára alkalmaztunk, statisztikai alapú, és felügyelt tanulást igényel. Minden egyértelműsíteni kívánt jelenséghez egy tanulókorpuszt kell konstruálni, amelyben kézzel megjelöljük, hogy a jelenség (többértelmű szóalak) egyes előfordulásai mely jelentésosztályba (mely lexémához) tartoznak. A szó környezetéből jegyeket vonunk el. A jegyek és a jelentésosztályok együttes előfordulásaiból egy tanuló algoritmus olyan modellt alkot, ami alapján a szó eddig nem látott előfordulásait jelentésosztályokba tudja sorolni. Tanulóalgoritmusnak a naiv Bayes-osztályozást és a döntési listás eljárást választottuk, amelyek a felügyelt szójelentés-egyértelműsítés népszerű megközelítései ([3]; [4], [5]).

3. A környezeti jegyek

A következő környezeti jellemzőket vettük figyelembe:

- tartalmas lemma jelenléte egy K méretű ablakban,
- szóalak jelenléte egy k méretű ablakban,
- inflexiós kategória jelenléte egy k méretű ablakban,
- szomszédos szóalakok,
- szomszédos szópárok (bigrammok),
- a többértelmű szó alakja.

Kétféle ablakméretet szoktak megkülönböztetni. A környezet által tartalmazott szemantikai információt egy szélesebb (mindkét irányban $K \approx 50$ méretű) ablakban keresik. Az itt található tartalmas szavak utalnak a kontextus témájára. Természetesen a szavak konkrét alakjától el kell vonatkoztatni, ezért kell a lemmákat figyelembe venni. Egy szűkebb (mindkét irányban $k \approx 3$ méretű) ablakban keresik a szintaktikai jellemzőket és az állandósult szókapcsolatokat. Mint a bevezetőben említettük, nem minden szintaktikai jellemzőt vett figyelembe a morfoszintaktikai egyértelműsítő, ezért érdemes a közeli inflexiós kategóriákat is felvenni a jegyek közé.

Az egyértelműsíteni kívánt szó alakját is érdemes a környezet jellemzőjének tekinteni. Ezt olyan homonímiák motiválják, amelyeknek tagjai nem pontosan ugyanúgy ragozódnak, vagyis egyes alakjaik egyáltalán nem homonimak. A *karja* alakból a *kar*[1] 'végtag' lexémára, a *kara* alakból a *kar*[2] testület' lexémára lehet következtetni, pedig mindkettő a *kar* FN lemma alakja. A morfológiai elemzőben hiába van meg ez a tudás, a kimenetből (lemma + morfológiai kategóriák) ez a megkülönböztetés eltűnik. Ezért ezeket a szabályokat ugyanúgy meg kell tanulnia az egyértelműsítő eljárásnak, mint a többi típusú jegyekből történő következtetést.

4. A felhasznált eljárások

4.1. A naiv Bayes-osztályozás

A naiv Bayes-osztályozás[3] azt az s jelentést rendeli egy C kontextusban megjelenő szóhoz, amelyre $P(s|C)$ maximális. Ennek kiszámításához felhasználja a Bayes-szabályt, miszerint $P(s|C) = \frac{(C|s)}{P(C)}P(s)$, és az úgynevezett Bayes-féle naiv feltevést, hogy a C környezet dekomponálható szavakra vagy egyéb jegyekre (f_j), amelyek egymástól független valószínűségi eseményeket alkotnak. Így a döntés a következőképpen írható le:

$$s = \arg\max_{s_i} P(s_i) \prod_{f_j} P(f_j|s_i)$$

A $P(f|s)$ valószínűségek a tanulókorpuszban található relatív gyakoriságok alapján becsülhetők. A döntésben minden jegy részt vesz, különböző mértékben.

4.2. Döntési listák

A döntési listákat [4] használta először jelentésegyértelműsítési célokra. Az eljárás kivonja a tanulóadatokból a jegyeket, és az alábbi formula alapján súlyokkal látja el őket:

$$w(s_i, f) = log\left(\frac{P(s_i|f)}{\sum_{j \neq i} P(s_j|f)}\right)$$

$s_1 \ldots s_n$ a lehetséges jelentések, f a jegy, $P(s|f)$ az s jelentés tanulóadatokban megfigyelt valószínűsége feltéve f jegy jelenlétét. A döntési eljárás a következő szerint hajtódik végre:

if f_i **then** s_k
else if f_j **then** s_l
...
else s_m

A feltételek vizsgálata a jelentés-jegy párok súlya szerint csökkenő sorrendben történik. Amennyiben nem születik döntés, a leggyakoribb jelentés kerül kiválasztásra. Látható, hogy az algoritmus minden esetben egyetlen, a legerősebbnek vélt jegy alapján hozza meg a döntést, mégis eredményesnek bizonyult.

5. A tanulókorpusz

A Értelmező Kéziszótár gyakorisági adatainak összeállításakor készült egy kézzel annotált korpusz. Ez olyan szóalakok konkordanciáit tartalmazza, amelyeket lefed a kéziszótár címszóállománya, és a jelen dolgozatban vizsgált többértelműségeket mutatják. Minden többértelműségi problémára (azaz homonimapárra – kar^1/kar^2 vagy lexémapárra – *sejt/sejtet*) készítettünk egy kétszáz elemű véletlen mintát az MNSZ-ből, és kézzel megjelöltük, hogy a többértelmű szóalakok mely lexémához tartoznak. Ezek közül 40 olyan homonímiát választottunk ki, amely kétszeres többértelműséget mutat, és amelynek mintájában mindkét jelentés képviselve van.

6. Kiértékelés

A tanulókorpuszon három algoritmust értékeltünk ki a 3. szakaszban meghatározott jegyekkel. Az első mindig a többértelműségen belüli relatív leggyakoribb jelentést rendelte az esetekhez, ez szolgált viszonyítási alapul. A másik kettő a fent említett Bayes-osztályozó és a döntési listák eljárás. A kiértékelést tízszeres keresztellenőrzéssel (cross-validation) végeztük: A tesztmintát tíz egyenlő részre osztottuk. A tanulás során mindig más részminta maradt ki a tanulómintából, a kiértékelés pedig a kimaradt részmintán zajlott. Az 1. táblázat bemutatja a kapott eredményeket. A három algoritmus által a különféle többértelműségeken elért pontosság rendre az Alap, Bayes és a DL oszlopban van felsorolva.

1. táblázat. A három eljárás pontossága a tesztadatokon – részletek

Lemma	Szófaj	Alap %	Bayes %	DL %
érem/érme	fn	86,43	87,86	86,43
gém	fn	77,78	95,56	94,44
kar	fn	51,67	87,22	75,00
karton	fn	91,00	91,00	91,00
méh	fn	54,44	83,33	83,33
passz	fn	61,18	91,76	92,94
prímás	fn	52,63	93,16	89,47
rák	fn	64,12	81,76	85,29
trópus	fn	85,26	87,37	86,84
hetes	mn	66,32	80,53	74,21
napos	mn	72,63	86,84	92,11
bán/bánik	ige	56,88	76,25	91,25
hajt	ige	78,46	80,00	80,00
megbíz/megbízik	ige	84,00	84,00	84,00
megtör/megtörik	ige	61,00	65,50	64,50
nyúl/nyúlik	ige	77,65	78,24	81,76
olt	ige	73,53	79,41	81,18
túl	hsz	71,11	83,34	83,56
Átlag a teljes mintán		*71,00*	*83,54*	*83,98*

A két vizsgált algoritmus átlagos pontossága közel egyforma, 83,54, illetve 83,98 százalékos volt, de vannak esetek, amikor az egyik sokkal gyengébben teljesít a másiknál (és viszont). Mindkettő minden esetben legalább olyan eredményt produkált, mint az alapalgoritmus, sőt általában jobbat. Minél nagyobb a többségi jelentés aránya, annál kisebb a Bayes- és a DL-algoritmus „előnye". Az alapalgoritmus pontossága a többségi jelentés relatív gyakoriságát közelíti. Amikor ez meghaladja a 80%-ot, a többi algoritmus nem tud szignifikánsan jobb eredményt elérni, azok is a többségi jelentést választják, mert a ritkábbik jelentésből kevés példa van a mintában.

Érdemes megvizsgálni, hogy a különböző típusú jegyek mekkora sikerrel vettek részt a döntési listákkal hozott döntésben. A *prímás* érte el a legjobb eredményt a viszonyítási alaphoz képest (70%-kal jobbat). A 2. táblázat mutatja a legerősebb jegyeket. Eszerint itt a bővebb környezet szavai adtak jobb támpontot az algoritmusnak. A 3. táblázat szerint viszont a *bánik/bán* eldöntésében leginkább a szóalak és a szűk környezete nyújtott segítséget. Az összes eset jegyeit megvizsgálva azt kapjuk, hogy várakozásunkkal ellentétben a szűk környezetben található esetkategóriák egyáltalán nem bizonyultak jelentésmegkülönböztető jegyeknek.

Az MNSZ-ben sok olyan homonim pár van, amelynek egyik tagja olyan ritka, hogy néhány százas véletlen mintában való előfordulásának nagyon kicsi az esélye (pl. *barát*[2] szerzetes', *fok*[1] 'eszköz tompa fele'). Kétséges, hogy volna értelme

a mintát addig növelni, amíg elegendő előfordulást nem kapunk. Ezeket a jelentéseket nem lehet statisztikai megközelítéssel felismerni.

Összefoglalva elmondhatjuk, hogy a naiv Bayes-osztályozás és a döntési listák eljárás hasonló pontossággal teljesített, és már relatív kis méretű tanulókorpuszon is jó eredményt kaptunk. A minta növelésével várhatóan még lehet javítani a pontosságon.

2. táblázat. A *prímás* példa tanult jegyei

Jegy	Súly	Döntés
bő k.: egyház	5.70	1
bő k.: érsek	5.67	1
bő k.: bíboros	5.67	1
bő k.: zenekar	4.94	2
...		

3. táblázat. A *bán/bánik* példa tanult jegyei

Jegy	Súly	Döntés
alak: bánom	5.25	bán
alak: bánnak	5.14	bánik
alak: bánja	4.78	bán
szűk k.: kell	4.70	bánik
szűk k.: vele	4.61	bánik
...		

Hivatkozások

1. Váradi, T.: The Hungarian National Corpus. In: Proceedings of the Second International Conference on Language Resources and Evaluation, Las Palmas (2002) 385–389
2. Oravecz, C., Dienes, P.: Efficient stochastic Part-of-Speech tagging for Hungarian. In: Proceedings of the Third International Conference on Language Resources and Evaluation, LREC2002, Las Palmas (2002) 710–717
3. Gale, W., Church, K., Yarowsky, D.: A method for disambiguating word senses in a large corpus. Computers and the Humanities 26 (1992) 415–439
4. Yarowsky, D.: Decision lists for lexical ambiguity resolution: Application to accent restoration in spanish and french. In: Meeting of the Association for Computational Linguistics. (1994) 88–95
5. Agirre, E., Martinez, D.: Exploring automatic word sense disambiguation with decision lists and the web (2000)

Corpus disambiguation – beyond the morphosyntax

Viktor Nagy

Research Institute for Linguistics, Budapest
nagyv@nytud.hu

Keywords: supervised statistical word sense disambiguation, decision lists

The Hungarian National Corpus is a large, morphological annotated corpus. Every word has an annotation which contains its lemma, part of speech and morphosyntactic category, produced by an automatic stochastic tagging procedure. This procedure can not distinguish phenomena having the same morphosyntactic distribution but different meaning because it treats the context as a sequence of morphosyntactic categories abstracted from the particular words. Such an analysis is often not adequate to determine which lexeme the word belongs to, because the (lemma, part of speech) pair can assign more than one lexeme to it if they are homographs with the same part of speech (according to the principles of Hungarian lexicography). While the morphosyntactic tagging can disambiguate the two meanings of word form *tűz*, namely *tűz*^{1} verb 'to pin' and *tűz*^{2} noun 'fire', it can not disambiguate the meanings of *ül* verb (1 'to sit', 2 'to celebrate') or *barát* noun (1 'friend', 2 'monk'). An additional problem arises when a word form belongs to different lemmas with the same syntactic behaviour. The word form *sejtette* can be the definite past singular 3th person form of *sejt* verb 'to guess' and *sejtet* verb 'to let guess', the *lappal* can be the singular instrumental form of *lap* noun 'sheet' and *lapp* adjective/noun 'Lapp'. In order to overcome the problem caused by the boundaries of morphosyntactic tagging, it is necessary to use a sort of word sense disambiguation.

Our approach uses a supervised learning algorithm based on a manually sense-tagged corpus. This corpus consists of sample concordances of the studied ambigous cases, about 200 occurrences for each case. The following context features was taken into account:

- bag of lemmas in a wider context,
- bag of word forms in a narrower context,
- bag of inflectional categories in a narrower context,
- form of the ambigous word.

We evaluated two learning algorithm on our data: the Naive Bayes method and the decision lists method. The studied examples were chosen to cover broad ranges of lexical ambiguity types. The two methods attained the same performance, they reached a precision of about 83% on the average.

Próbák és példák a Magyar értelmező kéziszótár (2. kiadás, 2003) rejtett információinak feltárására

Mártonfi Attila

MTA–ELTE Nagyszótári kutatócsoport
Budapest VI., Benczúr u. 33. 1068
rumci@nytud.hu

Kivonat. A tudásfeltárás, illetve ennek részeként az adatbányászat az információtechnológia divatos területei, melyek jellemzően az üzleti adatbázisok hasznosítására irányulnak, személete, eszköztára azonban – legalábbis részben – alkalmazható szótári adatbázisokra is. A VégSz.-ból nyomdatechnikai okok miatt kimaradt betűjegyben mért hosszúság, jelentésszám, etimológia, valamint szócikkfejben adott stílusminősítés pótlásánál az ÉKsz.[2] XML-változata segítségével teljesebb és korszerűbb adattábla hozható létre, ugyanis ez naprakész etimológiai információt nyújt a magyar szókészlet legtágabb köréről, valamint megtalálható benne a Magyar nemzeti szövegtár-beli abszolút gyakorisági érték is. Az így létrehozott relációs adatbázisból egyszerű lekérdezésekkel előállíthatók a különféle etimológiájú, lexikai minősítésű, szófajú vagy jelentésszámú szóhalmazok szótári, valamint – jelentős újdonságként – szöveggyakorisági mutatói. Az adatbányászat eszköztárával feltárhatók a fenti paraméterek közt fennálló rejtett mintázatok asszociációs szabályok kinyerése útján.
Kulcsszavak: lexikográfia, tudásfeltárás, etimológiai statisztika, Magyar értelmező kéziszótár

1. Bevezetés

A tudásfeltárás, illetve ennek részeként az adatbányászat az információtechnológia divatos területei, melyek jellemzően az üzleti adatbázisok hasznosítására irányulnak. Mivel azonban a cél – jelesül nagy adatbázisokból minél több rejtett adat, ismeretlen mintázat gépi úton történő kinyerése – lényegében a tudományos kutatás legáltalánosabb céljának tekinthető, így személete, eszköztára legalábbis részben alkalmazható szótári adatbázisokra is (a szótári adatbázisok mérete rendszerint nagyságrendekkel kisebb lévén az adatbányászat elsődleges területén előforduló hatalmas üzleti adatbázisoknál, a műveletek eszközigénye lényegesen kisebb, a kinyerhető információ azonban természetszerűleg szűkebb körű).

Az első jelentős magyar szótári adatbázis a VégSz. [3], illetve az ebből létrejött PC-s adatállomány (BUT). A szóvégszótár alapjául szolgáló adatbázis, a DT2 négymezőnyi többletet tartalmazott a papíron megjelent változathoz képest, ezek: a betűjegyben mért hosszúság, az ÉrtSz.-beli [2] jelentésszám, etimológia a SzófSz. [1] alapján, valamint az ÉrtSz. szócikkfejben adott stílusminősítése – ezek nyomdatechnikai okokból nem kerültek végül bele a papírszótárba, s így az ebből készült PC-s adatbázisba.

2. Az adatok átalakítása

A Magyar értelmező kéziszótár új kiadása [5], korszerű szótári munkálathoz méltó módon először XML-dokumentumként készült el, s bár grammatikai információi, melyek a VégSz. gerincét jelentik, lényegesen szegényesebbek, megfelelő átalakításokkal a fenti hiányok pótlásánál teljesebb és korszerűbb adattábla hozható létre. Korszerűbb, mert az ÉKsz.[2] naprakész etimológiai információt nyújt a magyar szókészlet legtágabb köréről, és teljesebb, ugyanis a szótár szófaji és lexikai minősítésén, valamint a kidolgozott jelentések számán kívül az adatbázisban minden címszónál megtalálható a Magyar nemzeti szövegtár-beli abszolút gyakorisági érték is, valamint gépi úton kódolható a fonémaszámban, illetve a szótagszámban mért szóhossz is.

Szükséges volt tehát az XML-dokumentumból egy olyan adattáblát létrehozni, mely egyszerű formában szolgáltatja a szükséges információkat, hogy a többé-kevésbé rejtett információk kinyerhetők legyenek. A konverziót követően (hiszen más struktúrában más adathibák tűnnek elő) el lehetett végezni néhány adattisztítási műveletet is. Az átalakított és megtisztított adattábla 72 444 rekordot tartalmazott, rekordonként 10 mezővel.

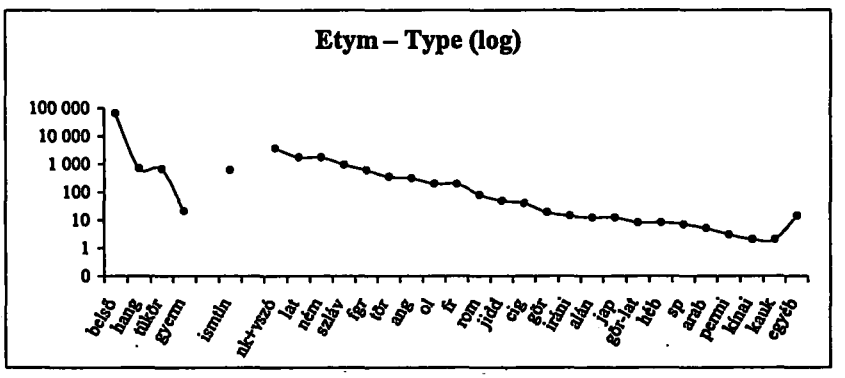
1. táblázat. Az adatbázis néhány rekordja

Azonosító	Lemma	Split	Syll	Phon	Freq	Usg	Pos	Sens	Etym
3053	asszonykerülő	☑	5	11	0		mn fn	1	
3054	asszonykéz	☑	3	8	21		fn	2	
3055	asszonykormány	☑	4	11	0	ritk tréf	fn	1	
3056	asszonymunka	☑	4	10	1	nép	fn	2	
3057	asszonynéni	☑	4	9	0	rég	fn	1	
3058	asszonynép	☑	3	8	62	nép	fn	1	
3059	asszonynév	☑	3	8	33		fn	1	
3060	asszonyos	☐	3	7	34		mn	2	
3061	asszonypajtás	☑	4	11	7	biz tréf	fn	1	
3062	asszonyrokon	☑	4	10	1		fn	1	
3063	asszonyság	☐	3	8	296		fn	3	
3064	asszonytárs	☐	3	9	15		fn	2	
3065	asztag	☐	2	5	92		fn	2	szláv
3066	asztal	☐	2	5	16627		fn	5	szláv
3067	asztalbontás	☑	4	11	4	vál	fn	1	

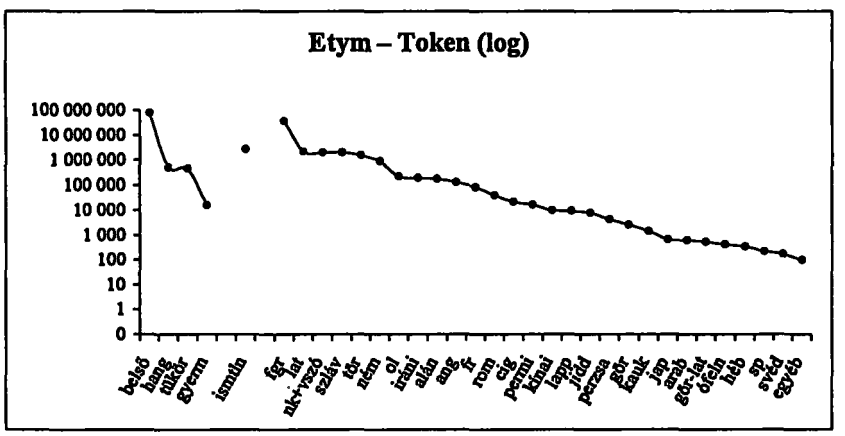
Az egyes mezők a következő információkat tartalmazzák: **Azonosító** – a rekord egyedi azonosítója; **Lemma** – a címszó főváltozata; **Split** – annak jelölője, hogy található-e | vagy ~ a címszóban (ennek megléte morfológiai tagoltságra utal, hiánya esetén lehet a szó morfológiailag tagolt vagy tagolatlan); **Syll** – a címszó szótagokban mért hossza; **Phon** – a címszó fonémákban mért hossza (a hosszú mássalhangzók kettőnek, a hosszú magánhangzók egynek számítanak); **Freq** – a Magyar nemzeti szövegtár-beli gyakorisági érték; **Usg** – a szócikkfejben szereplő lexikai minősítések, szóközzel elválasztva; **Pos** – a szófaji minősítések, szóközzel elválasztva; **Sens** – a jelentések száma; **Etym** – az etimológia, tömörített formában (lényegében az átadó nyelv vagy nyelvcsalád neve, esetenként a szó keletkezési módja).

3. Egyszerű lekérdezések

Az így létrehozott relációs adatbázisból egyszerű lekérdezésekkel elő lehetett állítani a különféle etimológiájú, lexikai minősítésű, szófajú vagy jelentésszámú szóhalmazok szótári, valamint szöveggyakorisági mutatóit (tekintettel azokra a mezőkre, amelyek több lexikai, illetve szófaji minősítést is tartalmaznak, n minősítés esetén a k-adik minősítés $\frac{k}{1+2+\mathrm{K}+n}$ pontot kapott – ez $n=k=1$ esetén szerencsére éppen 1)[1]. Efféle szöveggyakorisági mutatók korábban megfelelő adatbázis-, illetve korpuszháttér híján nem voltak számíthatók; a szótári gyakoriságok Papp Ferenc korábbi forrásokon alapuló hasonló vizsgálataival való összevetésre adnak lehetőséget ([3], [4]).



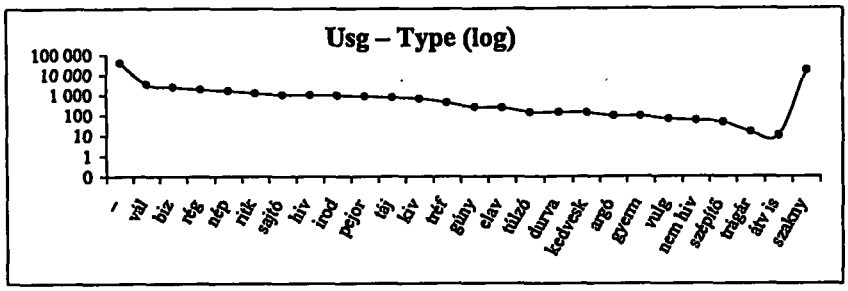
1. ábra. A különféle etimológiájú szavak szótári gyakorisága (logaritmikus skálán)[2]



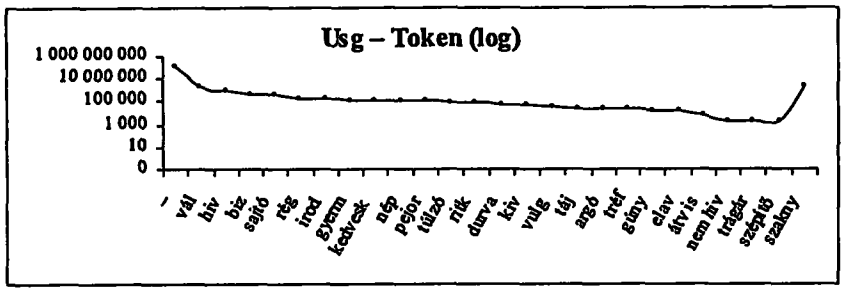
2. ábra. A különféle etimológiájú szavak szöveggyakorisága (logaritmikus skálán)

[1] A szöveggyakorisági értékek a számítás lényegéből adódóan pontatlanok, hiszen az egyes szövegszók szófajának vagy a lexikális minősítéseknek valódi eloszlásáról nincsen adatunk.

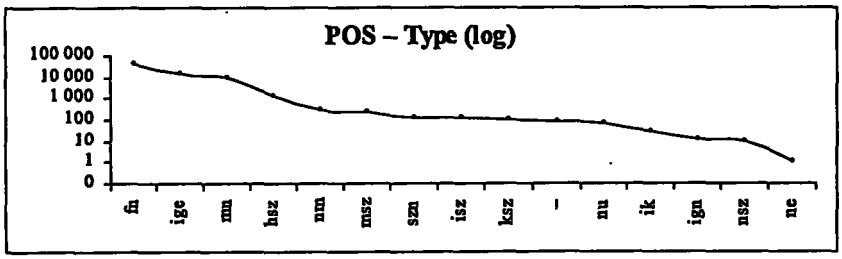
[2] Az etimológiai minősítés nélküli lemmák belső keletkezésűként számítottak.



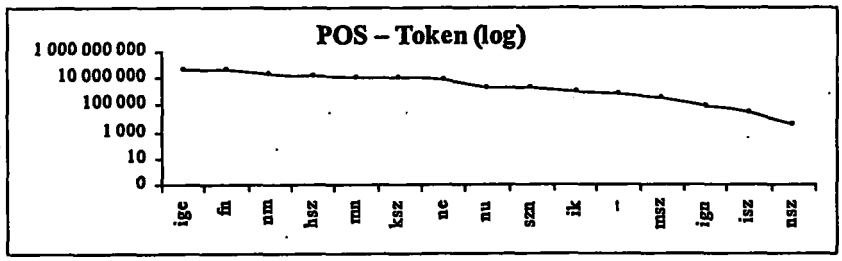
3. ábra. A különféle lexikai minősítésű szavak szótári gyakorisága (logaritmikus skálán)[3]



4. ábra. A különféle lexikai minősítésű szavak szöveggyakorisága (logaritmikus skálán)

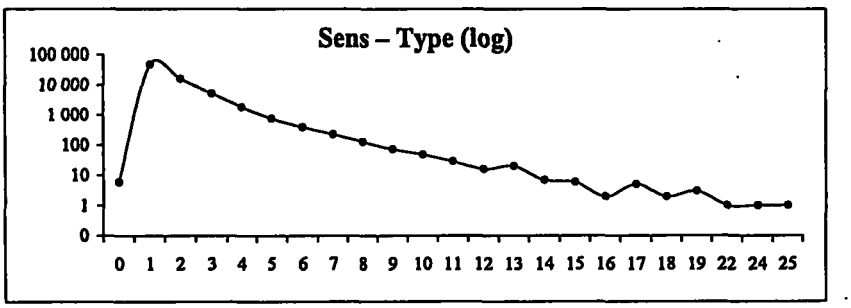


5. ábra. A különféle szófaji minősítésű szavak szótári gyakorisága (logaritmikus skálán)

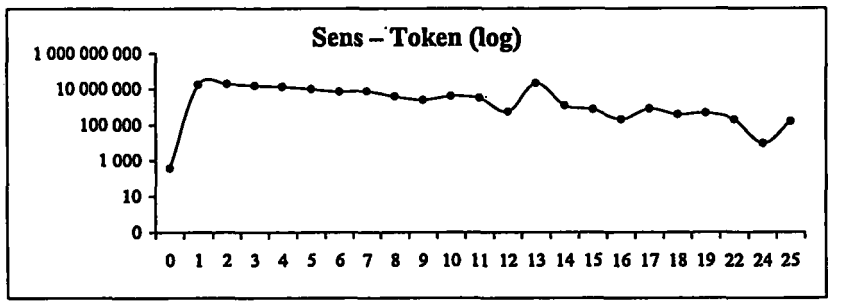


6. ábra. A különféle szófaji minősítésű szavak szöveggyakorisága (logaritmikus skálán)

[3] A szaknyelvi (tehát nagybetűs kezdetű) minősítéseket egységesen *szakny*-nak tekintettük.



7. ábra. A különféle jelentésszámú szavak szótári gyakorisága (logaritmikus skálán)



8. ábra. A különféle jelentésszámú szavak szöveggyakorisága (logaritmikus skálán)

4. Asszociációs szabályok

Az adatbányászat eszköztárával azonban – a fenti paraméterek közt fennálló rejtett mintázatok feltárását megcélzandó – ennél izgalmasabb elemzéseket is el lehetett végezni, asszociációs szabályok kinyerése segítségével. A rejtett mintázatok feltárásakor – több lexikai vagy szófaji minősítés esetén – csupán az első helyen állót vettük figyelembe.

Kiválogatva azokat a párosokat, amelyek tartója[4] legalább 10, összesen 7015 szabályjelölt állt elő, ezek közül a további vizsgálatok csupán az 1%-nál (tehát 724-nél) nagyobb tartójú 254 párt érintették, melyek közül csupán 103 volt legalább az egyik irányban legalább 67%-os valószínűségű szabály.

Azok az $A \rightarrow B$ asszociációs szabályok, amelyek esetében az A gyakorisága alacsony, ellenben a B gyakorisága magas, nem igazán beszédesek – habár ilyenből sok van. A legnagyobb gyakoriságú B-k a 'belső keletkezésű', illetve az 'egyjelentésű': a fenti 103 párból 71-et érintenek.

Bizonyos értelemben semmitmondó a fonémában, illetve szótagszámban mért szóhosszúság közti összefüggés, hiszen ez is triviális, beszédes is azonban, hiszen a magyar szótaghosszúságról állít valamit. E tárgyban a következő asszociációs szabályok adód-

[4] Tartó = az együttes előfordulás gyakorisága.

tak:[5] 5 fonéma ⇒ 2 szótag (8%; 92%); 3 fonéma ⇒ 1 szótag (1%; 87%); 7 fonéma ⇒ 3 szótag (11%; 81%); 10 fonéma ⇒ 4 szótag (9%; 79%); 8 fonéma ⇒ 3 szótag (12%; 77%); 4 fonéma ⇒ 2 szótag (2%; 73%); 13 fonéma ⇒ 5 szótag (2%; 68%); 12 fonéma ⇒ 5 szótag (3%; 67%). Egyiknek sem éri el a megfordítása az 50%-ot.

Habár a 'főnév' is igen nagy gyakoriságú, magas tartója és valószínűsége miatt érdemes megemlíteni a szakny ⇒ fn (20%; 87%) szabályt, illetve azokat a szabályokat, amelyek azt állítják, hogy a nagyon kis szöveggyakoriságú szavak jó eséllyel főnevek:[6] 0 db ⇒ fn (5%; 77%); 1 db ⇒ fn (3%; 72%); 2 db ⇒ fn (2%; 69%); 5 db ⇒ fn (1%; 68%); 6 db ⇒ fn (1%; 67%); 4 db ⇒ fn (1%; 67%). Megfigyelhető, hogy a hosszabb szavak szintén nagyobb valószínűséggel főnevek: 14 fonéma ⇒ fn (1%; 73%); 6 szótag ⇒ fn (3%; 71%); 13 fonéma ⇒ fn (2%; 71%); 12 fonéma ⇒ fn (4%; 70%); 5 szótag ⇒ fn (8%; 68%); 11 fonéma ⇒ fn (6%; 67%). A *rég* minősítésű szavak is igen gyakran főnevek: rég ⇒ fn (1%; 67%). Az etimológia szintén összefüggésben áll a főnévi szófajjal: szláv ⇒ fn (1%; 87%); nk+vszó ⇒ fn (4%; 81%); ném ⇒ fn (2%; 75%); lat ⇒ fn (2%; 70%) – ezen jelenség oka feltehetőleg az, hogy a főnév a legnyíltabb (tulajdonképpen az egyetlen teljesen nyílt) szóosztály, tehát a szókölcsönzések leginkább ezt érinti.

A fejben lexikálisan nem minősített lemmák[7] körében is megfigyelhetünk néhány szabályosságot. Természetes, hogy a többjelentésű címszavak szócikkfejében gyakran nincs lexikai minősítés: 5 jelentés ⇒ nincs lexikai minősítés (1%; 97%); 4 jelentés ⇒ n. lex. min. (2%; 93%); 3 jelentés ⇒ n. lex. min. (6%; 88%); 2 jelentés ⇒ n. lex. min. (17%; 77%). Összehasonlításképpen a monoszémák esetében: 1 jelentés ⇒ n. lex. min. (27%; 42%). Érdekes, szintén a lexikai minősítés hiányával kapcsolatos, nagy tartójú szabály: ige ⇒ n. lex. min. (16%; 71%) – melyet vélhetőleg az igék poliszemantikus hajlama indukál. Hasonló oka lehet az 1 szótag ⇒ n. lex. min. (2%; 67%) szabálynak.

5. Összefoglalás

A fentiekben áttekintettük, hogy melyek azok az információtípusok, amelyek rejtve maradnak egy szótárban, és bemutattunk néhány példát arra, hogy ezek miként nyerhetők ki.

Irodalom

1. Bárczi Géza: Magyar szófejtő szótár. Egyetemi Nyomda, Budapest (1941)
2. Bárczi Géza–Országh László (szerk.): A magyar nyelv értelmező szótára I–VII. Akadémiai Kiadó, Budapest (1959–1962)
3. Papp Ferenc (szerk.): A magyar nyelv szóvégmutató szótára. Akadémiai Kiadó, Budapest (1969)
4. Papp Ferenc: A debreceni thészaurusz. Linguistica. Series C. Relationes 11. MTA Nyelvtudományi Intézet, Budapest (2000)
5. Pusztai Ferenc (szerk.): Magyar értelmező kéziszótár. Akadémiai Kiadó, Budapest (2003[2])

[5] A zárójelen belül először a tartó, majd a szabály valószínűsége áll százalékban.

[6] A szabályok értelmezéséhez fontos tudni, hogy a főnevek adják a szócikkek 59%-át.

[7] Ezek a szócikkek 56%-át adják.

Attempts and examples for the discovery of hidden information of Concise explanatory dictionary of Hungarian (2nd edition, 2003)

Mártonfi Attila

MTA–ELTE Research Group of Academic Dictionary of Hungarian
rumci@nytud.hu

Keywords: lexicography, knowledge discovery, etymological statistics, Concise explanatory dictionary of Hungarian

Knowledge discovery and data mining – as its part – are trendy areas of IT, their aim is utilizing characteristically commercial databases. However the goal (namely extracting as much hidden data and unknown patterns as possible by machine) is essentially is the same as the most general goal of scientific research, therefore at least partially its approach and toolkit are applicable to lexicographical databases. (Since the size of lexicographical databases is usually smaller by orders of magnitude than monumental commercial databases occurring with the primer area of data mining, the device requirement of the operations is significantly less and the extractable information is more restricted.)

The first notable lexicographical database of Hungarian is Papp Ferenc's *Reverse-alphabetized dictionary of the Hungarian language* (VégSz.) and its derivative database on PC. The database which is the base of Papp's dictionary had four fields more than the paper-version: the length in characters, the number of senses in ÉrtSz. *(Explanatory dictionary of the Hungarian language),* the etymology based on *Etymological dictionary of Hungarian,* and the usage label given in the head of entries in ÉrtSz. – because of typographical reasons these are omitted from the paper-version and its derivative database.

The new edition of *Concise explanatory dictionary of Hungarian* (ÉKsz.[2]) – as an up-to-date lexicographical project should be – was first prepared as an XML document, and though its grammatical information (constituting the skeleton of VégSz.) is substantially more poor, with suitable conversions a more complete and more modern data tablet can be generated. It is more modern, because ÉKsz.[2] provides up-to-date etymological facts about the widest group of Hungarian words, and it is more complete, since apart from the part-of-speech and usage labels and the numbers of drawn senses all of the entries in this dictionary have the absolute frequency based on *Hungarian National Corpus,* furthermore the word-length in the number of phonemes or syllables can be coded.

With some simple queries the generated relational database gives token and type frequency indices of various etymology, usage label, part-of-speech or number of senses word-groups. Such token frequency indices – for want of a satisfactory database or

corpus background – formerly could not have been calculated; the type frequencies provide the possibility for comparision with Papp's examinations based on former sources.

With the toolkit of data mining more interesting analyses could be performed to discover hidden patterns of the above parameters by means of extracting association rules.

A magyar nyelv néhány szófaji elemzőjének összevetése

Kuba András[1], Bakota Tibor[1], Hócza András[1], Oravecz Csaba[2]

[1] Szegedi Tudományegyetem, Informatikai Tanszékcsoport
MTA-SZTE Mesterséges Intelligencia Kutatócsoport
andkuba@inf.u-szeged.hu, bakotat@math.u-szeged.hu, hocza@inf.u-szeged.hu
[2] MTA Nyelvtudományi Intézet
oravecz@nytud.hu

Kulcsszavak: szófaji egyértelműsítés, szabály alapú módszerek, Hidden Markov modell

Absztrakt. A dolgozatban három különböző POS tagger (szófaji egyértelműsítő) összehasonlítására vállalkozunk. Az első egy Hidden Markov Model alapú bigram elemző (*VMM*), a második egy szabály alapú módszer, amely bizonytalansági osztályok felhasználásával szófaji egyértelműsítést végez (*RGLearn*). Mindkét elemző a Szegedi Tudományegyetem Informatikai Tanszékcsoportján készült. A harmadik egyértelműsítő a jól ismert *TnT* [1], amely már több nyelven bizonyította képességeit, és amely a VMM-el szemben a szövegben előforduló szóhármasokat vizsgálja. Kísérleteinket a körülbelül 1,2 millió szót tartalmazó, kézzel annotált *Szeged Korpuszon* [2] végeztük, amely különböző szövegtípusokat foglal magába. Vizsgálatunk tárgya a szófaji egyértelműsítés, vagyis a mondatban előforduló adott szóra a lehetséges kódok közül a mondat szemantikáját visszatükröző egyértelmű tag meghatározása. Azaz a tesztelés során az egyes szavak bizonytalansági osztálya ismert volt az elemzők előtt. Ez alól a TnT kivétel, mivel ez a módszer a tesztelés során a szóvégződések elemzése által következtet az ismeretlen szavak lehetséges nyelvtani kódjára. A tesztelés során az RGLearn algoritmus 96,16% pontosságával megelőzte a VMM elemzőt (95,98%) illetve a TnT-t (95,08%). A hibásan taggelt szavak listájának összehasonlítása során kiderült, hogy a két statisztikai módszer "hajlamosabb" ugyanazokon a helyeken hibázni. A kapott eredményeket felhasználva, vizsgálatokat végeztünk arra nézve is, hogy a fenti módszereket kombinálva milyen találati pontosság érhető el.

1. Bevezető

Természetes nyelvi szövegek szófaji címkézése (*taggelése*) az egyik legalapvetőbb számítógépes nyelvészeti feladat. A nemzetközileg publikált módszerek közül a magyar nyelvre azonban csak néhányat ültettek át. [3] A legelterjedtebb eljárások közé a statisztikai illetve a szabály alapú módszerek tartoznak. A Szegedi Tudományegyetem, Informatikai Tanszékécsoportján több szófaji egyértelműsítő is kifejlesztésre került, ezeket hasonlítottuk össze más ismert módszerekkel több szempont szerint.

Hangsúlyozni szeretnénk, hogy a programok kizárólag szófaji egyértelműsítést végeznek, így az egyes szavak lehetséges kódjait a bemenettel együtt megkapják. Ennek a jelentős egyszerűsítésnek a fő oka, hogy ezeket a módszereket egy programlánc *(ToolChain)* [4,5] részeként használjuk, melynek egy korábbi fázisában a *HuMor* [6] morfológiai elemző előállítja az egyes szavak bizonytalansági osztályait. Ezzel biztosítjuk, hogy az egyértelműsítés során a módszerek nem találkoznak olyan szóval, amelynek nem ismertek a lehetséges szófaji besorolásai.

A legtöbb nemzetközileg ismert elemző, mint például a *TnT* is, saját beépített morfológiai elemzővel rendelkezik, mely általában a tanulás során előforduló szóvégződések, illetve a prefixek segítségével naiv következtetéseket tud levonni a szavak lehetséges tagjeit illetően. Az elemzők összehasonlítása során nem volt módunk arra, hogy a *TnT* előtt ismerté tegyük a szavak valódi bizonytalansági osztályát, így ez a módszer a többihez képest hátránnyal indult.

Kísérleteinket a körülbelül 1,2 millió szót tartalmazó, kézzel annotált Szeged Korpuszon végeztük, amely különböző szövegtípusokat foglal magába. A korpusz nagyon részletes MSD kódolást használ, az egyes tagek jelölésére, így az előforduló különböző tagek száma meghaladja az 1400-at.

A következő fejezetben a *VMM, TnT* illetve az *RGLearn* taggerek jellegzetességeit ismertetjük.

2. A VMM tagger

A VMM valójában egy Hidden Markov Modellt megvalósító algoritmus. [7] A tanulás során a Modell paraméterei közvetlenül számíthatók, mivel a tréning alapjául szolgáló korpuszban a szavak helyes szófaji kódjai be vannak jelölve. Más szavakkal a modellben az állapotátmenetek ismertek a tanulás ideje alatt, éppen ezért *Visible Markov Model* néven is emlegetik. A tréning során nem csak a modell paraméterei kerülnek kiszámításra, hanem statisztikák készülnek az egyes bizonytalansági osztályokra is: melyik osztály hányszor fordult elő, melyik volt a leggyakrabban kiválasztott szófaji kód, és az hányszor bizonyult helyesnek. Ha a tesztelés során olyan szót vizsgálunk, amely nem fordult elő a tanítás során, akkor az illető szó bizonytalansági osztályában előforduló kódokhoz kezdeti eloszlást rendelünk, amely megfelel a korpuszból előzetesen kigyűjtött adatoknak. Az egyértelműsítő módszer a számára ismert szavakhoz azt a kód-eloszlást rendeli, amely az adott szóra a tanulás során kialakult. Az egyértelműsítés a Viterbi algoritmus [7] segítségével történik.

A többféle tesztelés során három szinten mértük az egyértelműsítő módszer találati pontosságát:

1. szint: a módszer által eredményül adott, illetve a korpuszban kézzel meghatározott kódok első karakterének egyezése – szófaj meghatározás,
2. szint: összevont kódrendeszer – az egymástól lényegesen nem különböző szófaji címkéket csoportokba rendeztük, és a csoporton belüli címkéket azonosnak vettük, azaz itt nem követeltünk meg teljes egyezést
3. szint: az *MSD* kódok teljes egyezése

A találati pontosságot nem csak az összes szó arányában, hanem a többértelmű szavak tekintetében is megvizsgáltuk. Ugyanis hibás döntést csak a többértelmű szavakon hozhat az algoritmus (ahol egy szónak több különböző szófaji besorolása is lehetséges).

Vizsgáltuk többek közt a tagger átlagos viselkedését (*90-10 cross fold validation*), melynek az eredményeit az alábbi táblázatban foglaltuk össze:

Elfogadási szint	Min	Max	Eltérés	Átlag
1. szint	97.11%	97.67%	0.56%	97.48%
2. szint	95.74%	96.38%	0.64%	96.16%
3. szint összes szóra vetítve	95.52%	96.17%	0.65%	95.93%
3. szint a többértelmű szavakra vetítve	90.26%	91.69%	1.43%	91.20%

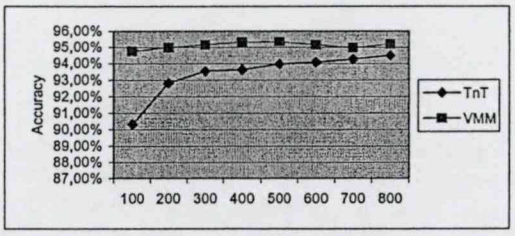
1. Táblázat: 90-10 cross-fold validáció eredményei az egyes szintekre lebontva

3. TnT (Trigrams'n'Tags)

A *TnT* szintén statisztikai elven működő szófaji egyértelműsítő program, amelyet *Thorsten Brants (Saarland University)* fejlesztett ki a 90-es évek elején. [1] Előnye, hogy tetszőleges nyelvre alkalmazható, nagy találati pontosságot képes elérni, és gyors. Hátránya azonban, hogy az egyes szavak bizonytalansági osztálya nem adható meg közvetlenül, hanem saját beépített morfológiai elemzőjével igyekszik meghatározni a lehetséges szófaji címkéket. Ha a szófaji besorolás nem túl részletes, azaz nincs sok

lehetséges címke, akkor nagyon jó eredményeket produkál (96-97% per word), azonban a mi vizsgálatainkban 1-1,5%-kal lemarad a többi taggertől a találati pontosságot tekintve.

A tesztelések során vizsgáltuk a TnT találati pontosságát, illetve, hogy az hogyan változik a tréninghalmaz növelésével. Az eredményeket összehasonlítottuk a VMM – nél kapott adatokkal:



1. Ábra: A találati pontosság alakulása a tréning méretének növelésével a TnT, illetve a VMM elemzők esetében. A tréning mérete a korpusz fájlok számával van megadva.

4. Szabály alapú taggerek

A szabály alapú módszereknek számos előnyük van a statisztikai taggerekhez képest:

- A szabályok könnyen áttekinthetők, értelmezhetők,
- Könnyen kiegészíthetők szakértői tudás beépítésével, amelyek megnyilvánulhatnak szakértők által adott szabályokban, kezdeti hipotézisben, vagy a meglévő szabályok finomításában
- A támogatás gyors és egyszerűen megvalósítható.

4.1 Az RGLearn szabályrendszer tanuló algoritmus

Az RGLearn egy saját fejlesztésű algoritmus, amely egy kezdeti szabályrendszerből kiindulva azt úgy próbálja általánosítani, hogy a tréning példákat a lehető legkevesebb számú, minél általánosabb szabállyal lefedje, úgy hogy a szabályok hibája (amikor rossz döntést hoznak) egy adott küszöbérték alatt maradjon. A kezdeti szabályrendszer lehet más tanuló módszerek vagy nyelvész szakértők által előállított szabályrendszer is. Az általunk használt kezdeti szabályrendszer a nem alapértelmezett választást tartalmazó tréning példákat tartalmazta. Alapértelmezett választás az a szófaji kód, amely a leggyakrabban előfordul az adott szóra nézve.

```
RULE_SET = non default cases from EXAMPLE_SET
while change RULE_SET do
{
     foreach RULE of RULE_SET do unification RULE
     foreach RULE of RULE_SET do generalization RULE
     foreach RULE of RULE_SET do delete rules covered by RULE
}
```

Unification RULE:

1.) Megkeresi azt a szabályt ami a RULE szabályhoz legjobban hasonlít (az attribútumok értékei (szavak, nyelvtani kódok) a legtöbb karakter pozíción egyeznek az értékek elejétől).
2.) A két szabályt összevonja (a különböző részeket elhagyja)
3.) A két szabály helyett bevezeti az összevont szabályt ha annak pontossága nagyobb egy előre megadott küszöbértéknél.

Generalization RULE:

1.) Készít egy új szabályt RULE szabályból úgy, hogy a környezet szélétől befelé haladva egy attribútum értékét általánosabb reguláris kifejezésre cseréli vagy elhagyja.

3.) A RULE szabály helyett bevezeti az általánosabb szabályt ha annak pontossága nagyobb egy előre megadott küszöbértéknél.

4.2. A szabály alapú szófaji egyértelműsítő működése

A szófaji egyértelműsítő által használt szabályrendszer többféle szempont szerint rendezett. Ezáltal gyors (bináris) keresés valósítható meg a döntéshez szükséges rész-szabályhalmaz kiválasztásához. A szabályok kiválasztásakor a sok példát lefedő, minél pontosabb szabályokat próbáljuk alkalmazni először. Az egyértelműsítés mondatonként történik, esetenként több menetben amíg a tagger talál új egyértelműsíthető szót. Ha egy adott többértelmű szóra nincs szabály az az alapértelmezett (leggyakoribb) nyelvtani kódot kapja meg a bizonytalansági osztály választási lehetőségei közül. A szófaji egyértelműsítés algoritmusa a következő:

```
while change do
{
      foreach tag of sentence do
      if tag not decided then
      foreach rule of ruleset do
      if rule covers tag then decide code of tag by using rule
}
foreach tag of sentence do
if tag not decided then set default code of tag
```

4.3. A C4.5 döntésifa tanuló algoritmus

A C4.5 algoritmus az ID3-algoritmus egy továbbfejlesztett változata. J. R. Quinlan nevéhez fűződik. [8] A C4.5 egy döntési fát állít elő, melyben a csomópontok egy-egy attribútumra vonatkozó kérdések, a levelek pedig a döntések. A C4.5 úgy próbálja előállítani a döntési fát, hogy minél kevesebb kérdéssel el lehessen jutni a döntéshez, ezért azokat az attribútumokat választja ki csomópontoknak, melyeknek legnagyobb az információs nyeresége. A döntési fa pedig átkonvertálható szabályokká.

5. Eredmények

Az alábbi táblázat a 4 tárgyalt egyértelműsítő módszer, és referenciaként a jól ismert C4.5 módszer által elért eredményeket mutatja egy konkrét teszt esetén.

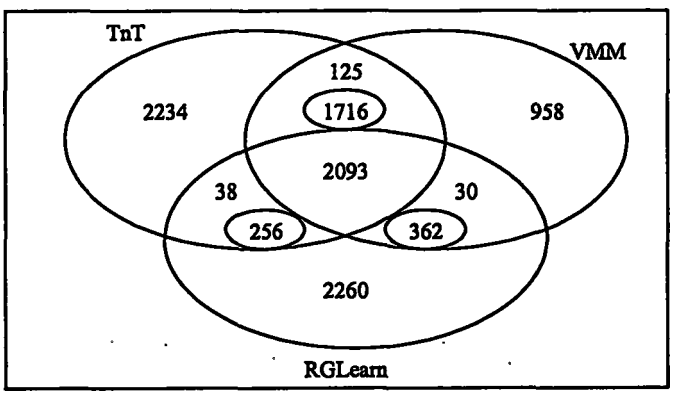
	VMM	TnT	C4.5	RGLearn
Tréning idő	10 perc	30 mp	~6 óra	~24 óra
Teszt idő	11 perc	30 mp	3 perc	2 perc
Tréning fájlok száma	823	823	823	823
Teszt fájlok száma	91	91	91	91
Szavak száma	131345	131345	131345	131345
Többértelmű szavak száma	60758	60758	60758	60758
Hibásan jelölt szavak száma	5284	6462	6646	5039
Hibásan jelölt szavak (1. szint)	3435	-	4338	3043
Pontosság				
3. szint az összes szón	95,98%	95,08%	94,94%	96,16%
3. szint a többértelmű szavakon	91,30%	89,36%	89,06%	91,71%
1. szint az összes szón	97,38%	-	96,70%	97,68%
1. szint a többértelmű szavakon	94,35%	-	92,86%	94,99%

2. Táblázat: A három egyértelműsítő módszer által ugyanazon a tréning és teszt adaton produkált eredmények

A módszerek közül az RGLearn érte el a legjobb eredményt. Figyelemre méltó, hogy az alapvetően más megközelítéssel dolgozó módszerek mennyire hasonló eredményeket produkálnak. Ha a teljes egyezés helyett megelégszünk az első szintű egyezéssel, akkor minden módszer eredménye nagyjából 1,5 százalékot javul. Nagy különbség van azonban az egyes módszerek tanulási és futási időigénye között. Erre feltétlenül figyelemmel kell lenni, ha a módszereket alkalmazni kívánjuk.

5.1. Kombinált módszerek

A következőkben csak a három legjobb eredményt produkáló egyértelműsítő algoritmusra koncentrálunk. Az algoritmusok nem csak pontosságban, hanem az elkövetett hibák típusában is különböznek. Az egyes módszerek esetében érdemes összevetni a helytelenül taggelt szavak listáját. Az alábbi ábrán látható a hibásan megjelölt szavak száma a fenti teszt esetében:



2. Ábra: Az egyes módszerek által hibásan megjelölt szavak eloszlása. Az egyes halmazok ábrázolják az adott módszer által hibásan jelölt szavakat. A metszetekben (ahol egy módszer helyes, kettő pedig helytelen eredményt ad) külön csoportosítottuk azokat a szavakat, amelyekre a két hibázó módszer ugyanazt a (hibás) eredményt adta.

A kérdés, hogy lehet-e a legpontosabb tagger algoritmusnál is jobb eredményt elérni egy kombinált módszer alkalmazásával, amely figyelembe veszi a kevésbé pontos algoritmusok eredményeit is?

Ahol a 3 halmaz metszi egymást, azaz mindhárom tagger hibás eredményt ad, ott egy kombinált módszer sem segíthet, de azon szavak esetében, ahol legalább az egyik tagger jól dönt, van esély a helyes megoldás megtalálására. A kialakuló kombinált módszer hatékonysága nyilván a döntési stratégián múlik. A döntési stratégia mondja meg, hogy ha egy szó esetén mindhárom módszer által adott eredményt ismerjük, ezek közül melyiket válasszuk. Mi olyan döntési stratégiákat vizsgáltunk, amelyek függetlenek az egyes módszerek által választott tagektől.

Ha a 3 módszer 3 különböző választ ad, akkor valamelyik módszert ki kell tűntetnünk, és a preferált módszer által szolgáltatott eredményt fogadjuk el véglegesnek. Az ábráról leolvasható, hogy ilyen esetben az RGLearn algoritmust érdemes preferálni, mivel ez 125 esetben ad helyes választ a másik két módszer 30, illetve 38 találatával szemben.

Ha a módszerek által szolgáltatott tagek közül kettő megegyezik, de a harmadik eltér ettől, akkor alapvetően két dolgot tehetünk: vagy az egyező taget választjuk, vagy a különbözőt. Az ábráról leolvasható, hogy (nem meglepő módon) ilyenkor azt a taget érdemes választani, amelyiket két módszer egyformán eredményül adta.

Ha a módszerek által adott mindhárom tag megegyezik, nyilván nincs módunkban mást tenni, mint ezt a taget eredményül adni.

A fentiek értelmében tehát az elérhető legjobb algoritmus a korábban látott teszt esetén az, hogy a három módszert megszavaztatjuk: mindhárom módszer eredményének ismeretében azt a taget választjuk. Amelyre legalább két módszer szavazott, ha pedig nincs ilyen, akkor az RGLearn módszer által adott eredményt választjuk. Ezzel a stratégiával az elérhető pontosság az összes szóra vetítve 96,58%-nak, a többértelmű szavakon 92,6%-nak adódik. Azaz a pontosság a többértelmű szavakon a legjobb módszerhez képest is kb. 1 százalékot javult.

6. Összegzés

A Szeged Korpusz jó alapot ad ahhoz, hogy a különböző szófaji egyértelműsítő módszereket összehasonlíthassuk. A négy vizsgált módszerből (a C4.5 algoritmust is ideértve) kettő szabályalapú, kettő pedig statisztikai volt; mindkét csoportból volt egy standard, jól bevált eszköz (C4.5 és TnT) valamint egy általunk megvalósított módszer (VMM és RGLearn). A módszerek nagyjából hasonló eredményt produkálnak, a legjobb eredményt az RGLearn algoritmus adta.

Vizsgáltuk azt is, hogy nagyobb szövegen javulna-e a módszerek pontossága. A tréning adatok további növelésével lényegesen jobb eredményt már nem várhatunk.

A pontosság tovább növelhető viszont, ha a rendelkezésre álló módszereket párhuzamosan felhasználjuk valamilyen kombinált módszerben. Egy ilyen algoritmussal az egyértelműsítés pontosságát kb. 1 százalékkal javítani tudtuk a legjobb algoritmushoz képest.

A kutatás eredménye tehát egy viszonylag jó pontosságú szófaji egyértelműsítő program lett, amely bármilyen magyar nyelvű korpusz annotálására használható. Ez a jövőben egy olyan rendszer modulja lesz, amely magyar nyelvű természetes szöveget dolgoz fel információ-kinyerés céljából. [4,5]

Az automatikus módszerek a kézzel annotált korpuszok hibajavításában is segíthetnek. Jelenleg a Szeged Korpusz munkálataiban hibajavításra használjuk az itt ismertetett módszereket. Azokat a szavakat, ahol az automatikus módszer hibázik, nyelvész szakértők átnézik, és amennyiben szükséges, javítják. A cikk írásáig körülbelül 8000 hibásan annotált szóra derült így fény. Ez az összes szó kb. 0,7 százaléka.

Irodalom

1. Brants, T.: *TnT – A Statistical Part-of-Speech Tagger*, Saarland University, Computational Linguistics (2000)
2. Alexin, Z., Csirik, J., Gyimóthy, T., Bibok, K., Hatvani, Cs., Prószéky, G., Tihanyi, L. (2003) *Manually Annotated Hungarian Corpus*, in Proceedings of the Research Note Sessions of the 10th Conference of the European Chapter of the Association for Computational Linguistics EACL03, Budapest, Hungary, pp. 53–56.
3. Horváth, T., Alexin, Z, Gyimóthy, T. and Wrobel, S. *Application of Different Learning Methods to Hungarian Part-of-speech Tagging*, in Proceedings of 9th International Workshop on Inductive Logic Programming (ILP99) Bled, Slovenia, in the LNAI series Vol 1634 p. 128–139, Springer Verlag (1999)
4. Hócza, A., Alexin, Z., Csendes, D., Csirik, J., Gyimóthy, T.: *Application of ILP methods in different natural language processing phases for information extraction from Hungarian texts* in Proc. of the Kalmár Workshop on Logic and Computer Science, Szeged, Hungary, 1-2 October, pp. 107-116 (2003)
5. Freitag D. *Machine Learning for Information Extraction in Informal Domains*, Machine Learning, 39, 169–202. (2000)
6. Prószéky, Gábor: *Humor: a Morphological System for Corpus Analysis*, Language Resources for Language *Technology* (Proceedings of the First European TELRI Seminar), 149-158. Tihany, Hungary (1995)
7. Manning, C. D., Schütze, H.: *Foundations of Statistical Natural Language Processing*, Chapter 9. MIT Press (1999)
8. Quinlan, J. R. *C 4.5: Programs for Machine Learning*, Morgan Kaufmann Publisher. (1993)

Comparing different POS-tagging techniques for Hungarian

András Kuba[1], Tibor Bakota[1], András Hócza[1], Csaba Oravecz[2]

[1]University of Szeged, Department of Informatics
Research Group on Artificial Intelligence
andkuba@inf.u-szeged.hu, bakotat@math.u-szeged.hu, hocza@inf.u-szeged.hu
[2]Research Institute for Linguistics at the Hungarian Academy of Sciences
oravecz@nytud.hu

Keywords: POS tagging, rule-based methods, Hidden Markov Model

Abtsract

In recent years, many different techniques have been developed for tagging natural languages, but only a few of them were implemented for Hungarian. The most commonly used types are statistics based taggers, but also the rule based ones are very widespread. We currently took up the challenge to compare four taggers. The first one, *VMM*, has been developed at the University of Szeged, Department of Informatics, and it is based on Hidden Markov Model method. The second one, also developed at the Department of Informatics is a rule based tagger called *RGLearn*. These two taggers were compared with the well-known TnT software, and the C4.5 algorithm. TnT is also a statistics based tagger, but it operates with trigrams, while *VMM* is just a bigram tagger. The comparison was performed on the so-called Szeged Corpus, which is a manually annotated set of text containing approx. 1.2 million words from different topic areas. Because very fine MSD encoding was used (there are approx. 1 500 different tags in the MSD encoding system), the taggers aren't expected to perform well. In our case POS tagging is the problem where the ambiguity class of each word is known, and the tagger decides which tag sequence represents the correct meaning of the sentence. The reason of this assumption is that these taggers should be part of a *ToolChain*, where in an earlier phase the *HuMor* morphological tagging software generates the possible tags of each word. In this sense *RGLearn* was found to be the best, performing 96,16% per word accuracy, *VMM* performed 95,98%, TnT 95,08%, and finally C4.5 94,94%. Here we have to note that information about ambiguity classes of words were not available for TnT. During the training TnT generates a suffix tree, which is used for a naive morphological examination of each word in order to determine its possible tags. This heuristics rapidly increases the accuracy on unknown words. After these tests were finished, a list of mistaken words was extracted for each tagger, and than compared. We've found that *VMM* and *TnT* had more mistakes in common than any other two taggers. This is due to the fact that both taggers use statistical methods. Using these results, we've come to conclusions about how these taggers could be combined in order to produce better results. The list of mistakes was forwarded to linguists for analysis, and to find out weather the machine or human made the mistake. The results also are used to make corrections to the corpus. Till now some 8000 mistakes were noticed and corrected by using this method, which is about 0.7% of the whole corpus.

Szótípusok bevezetésének szabályszerűsége magyar és angol nyelvű nyomtatott szövegekben

Csernoch Mária[1], Hunyadi László[2]

[1]Angol-Amerikai Intézet,
[2]Általános és Alkalmazott Nyelvészeti Tanszék
4010 Debrecen, Hungary
mcsernoch@hotmail.com

Absztrakt. Magyar és angol nyelvű irodalmi művekben, valamint angol nyelvkönyvekben vizsgáltuk a szótípusok megjelenésének szabályszerűségét. A szövegek egyenlő hosszúságú, rövid intervallumokra való darabolásával a szövegben bekövetkező apró változások is nyomon követhetővé váltak. Az eredeti mű szótípusainak gyakorisága alapján elkészült egy modell, egy mesterséges szöveg. Az eredeti és a mesterséges szöveget összehasonlítva meg tudjuk határozni azokat a szövegrészeket, amelyek kiugranak a történetből, nem épülnek be szorosan az események folyamatába. A nyelvkönyvek vizsgálata során azt tapasztaltuk, hogy egy nyelvkönyv szókészlete sokkal inkább véletlenszerűen összeválogatott, majd összefűzött novellák szókészletéhez hasonlít. Megfigyeléseink szerint a nyelvkönyvek tervezésénél a szerzők figyelmen kívül hagyják, hogy nemcsak a szótípusok számát kellene meghatározni és magasan tartani, hanem az egyes típusok megfelelő számú ismétléséről is gondoskodni kellene.

Bevezetés

Szavak gyakoriságán alapuló modellek feltételezik, hogy a szavak véletlenszerűen jelennek meg egy műben. Azonban a véletlenszerű válogatások is több különböző stratégia alapján végezhetőek el [1], [2], [3]. A legjobb modelleket azok a kísérletek adták, amelyek feltételezték, hogy a típusok binomiális eloszlást követnek. A típusok binomiális eloszlását feltételezve Baayen korábbi [3], majd Hoover által megismételt [4] modellje minden m-szer előforduló szótípushoz kiszámított egy konstanst, amely az m-szer előforduló szótípusok várható száma lesz a szöveg tetszőleges pontján. Elvégezve az összegzést a szövegben előforduló valamennyi szótípusra, meghatározható az adott szövegrészben előforduló szótípusok száma. Az így előállított modell, következésképpen statikus. Egy adott M ($M \leq N$, ahol N = a szövegben előforduló szavak száma) esetén mindig azonos. A nagy teljesítményű személyi számítógépek megjelenésével azonban ma már lehetőség van dinamikus modellek építésére is.
A kísérletek fő célja az volt, hogy egy ilyen dinamikus modell megalkotásával meg tudjuk mutatni, milyen okokkal magyarázható egy új szótípus megjelenése, melyek azok a források, amelyek az írót egy új szótípus bevezetésére ösztönzik.

Módszerek

Baayen modelljében a szövegek 40 egyenlő hosszúságú intervallumra darabolódnak, s így 40 különböző mérési pontban lehet elvégezni a számolást és ábrázolni a kapott eredményeket [2], [3]. Az itt bemutatásra kerülő modell szintén az eredeti szöveg típusainak gyakoriságát használja kiindulási pontként, de folytatásként a típusok relatív gyakorisága és előfordulási valószínűsége kerül kiszámolásra, majd ezen értékek ismeretében előállítunk egy eloszlásfüggvényt. Az eloszlásfüggvény értékkészletének elemeit véletlenszerűen válogatjuk, majd a függvény alapján visszakereshető az értékkészletnek az az eleme, amelyhez a véletlenszerűen választott szám hozzá lett rendelve. A véletlen szám válogatását mindaddig ismételjük, amíg el nem érjük az eredeti szószámot. Ezzel az eljárással elő tudunk állítani mesterséges szövegeket, amelyek az eredeti mű típusainak gyakoriságából származtathatóak.
Modellünk abban is eltér az előzőektől, hogy a szövegeket nem konstans számú darabra osztja függetlenül a szöveg hosszától, hanem a blokkok hossza lesz állandó. Általában 100 szó hosszúságú blokkokat használtunk, és ennek megfelelően a blokkok száma változó volt. Az így elvégzett darabolásnak két előnyét is találtuk a régiekkel szemben. Az egyik, hogy a blokkok hosszúsága független a szöveg hosszától, így a különböző hosszúságú szövegek darabkái sokkal inkább összevethetőek, mint különböző hosszúságú blokkok esetén. A másik előny a rövidre választott blokkhosszúságból ered. Rövid blokkokat használva a szövegben bekövetkező apró változások is nyomon követhetőek (1. ábra).

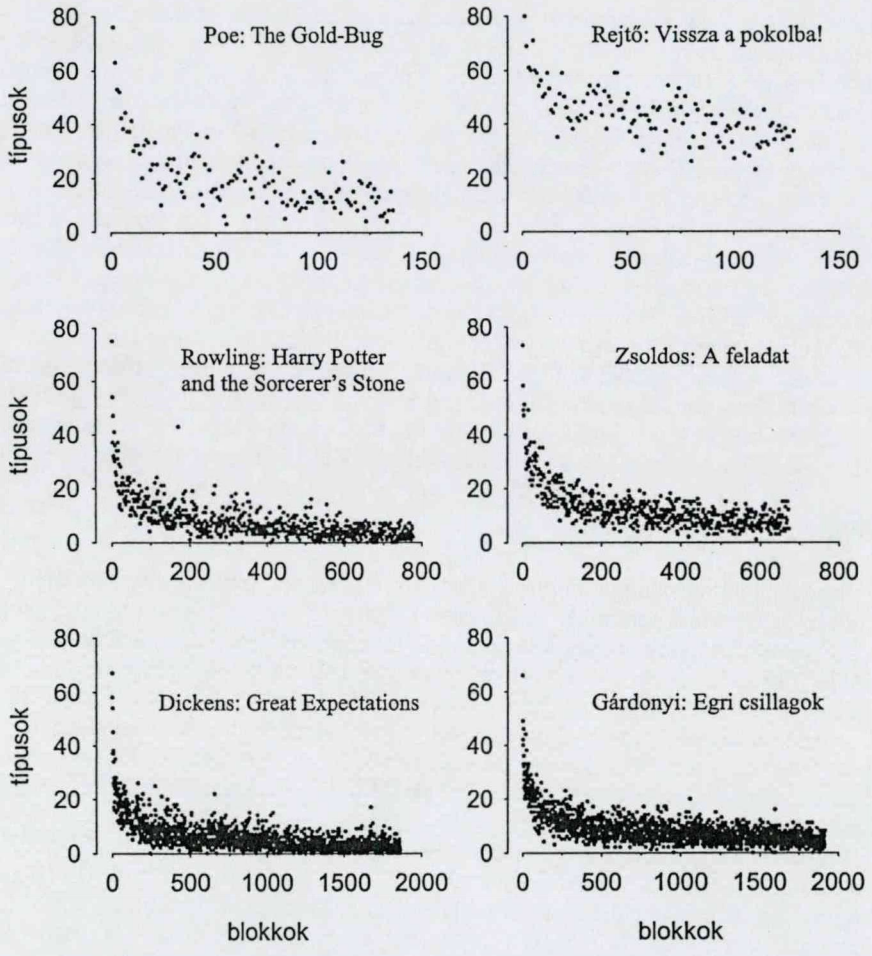
Felhasznált anyagok

A rövid blokkra bontott szöveg ismét felveti azt a kérdést, hogy a típusok gyakorisága, mint paraméter alkalmas-e a szerző azonosításra. Ahhoz, hogy összehasonlítható eredményeket kapjunk, a szövegek kiválasztása az alábbi stratégia alapján történt: egy szerző több műve, egyszerzős sorozat kötetei, egyazon műfajhoz sorolható művek különböző szerzőktől, összefűzött novellák egy illetve több szerzőtől, egynyelvű nyelvkönyvek. A program alapértelmezés szerint angol és magyar nyelvű szövegek feldolgozására alkalmas, de a felhasználónak lehetősége van saját karakterkészletének beállítására, így további, más nyelvű művek feldolgozására is alkalmassá tehető.
A szövegek feldolgozásához az eredeti, nyomtatott szövegek elektronikus verziójára volt szükség. Az elektronikus szövegek fő forrása az Internet volt, az Interneten ingyenesen nem elérhető szövegek pótlása kézi szkenneléssel történt. A különböző forrásból származó szövegek egységesítését, szabványosítását a szövegek feldolgozása előtt meg kellett oldani. Itt szeretnénk megjegyezni, hogy a feldolgozandó szövegek elérhetősége is nagyban befolyásolta, hogy melyeken végeztük el kísérleteinket.

Eredmények

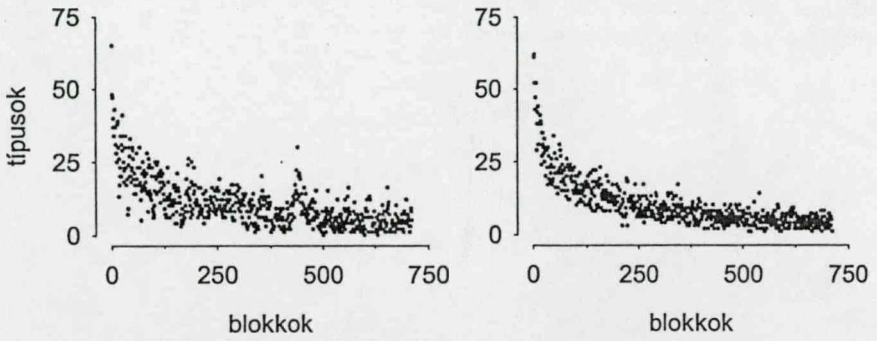
A program a vizsgált szövegeket többféle szempont alapján is elemzi. A gyakoriságok szolgáltak a modell kiindulási értékeiként. A gyakoriságok ezen túlmenően

alkalmasak arra, hogy összevessük őket nagy korpuszon végzett gyakorisági vizsgálatokkal. Ennek az összehasonlításnak igazán nagy jelentősége a nyelvkönyvek szókészletének vizsgálatánál van. A nyelvkönyvek szókészletéről valóban reális képet azonban akkor kapnánk, ha rendelkezésünkre állna egy szókészlet minimum és ezzel lehetne összehasonlítani a program által kiszámolt értékeket.



1. ábra. Angol (balra) és magyar (jobbra) szövegek szótípusainak megjelenése különböző hosszúságú szövegek esetén. A szövegeket 100 szó hosszúságú blokkokra daraboltuk. Az ábrák az újonnan megjelenő típusok számát mutatják. Fent „rövid", kb. 15000; középen „közepes", kb. 80000; lent „hosszú", kb. 200000 szó hosszú szövegek elemzésének eredménye látható

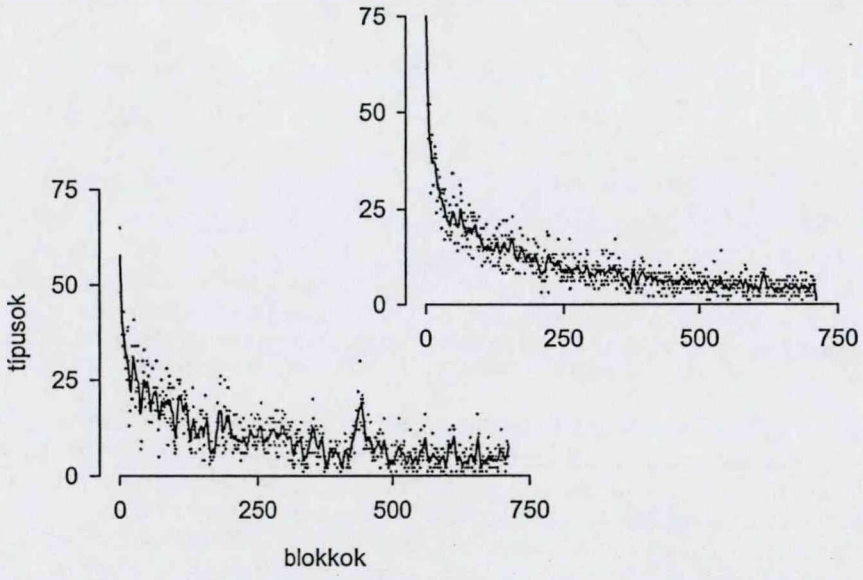
A program ábrázolja a blokkban megjelenő új szavak számát. Látványosan olyan pontok ugranak ki a függvény menetében, amelyek a történethez szervesen nem kötődő esemény bekövetkezésére utalnak.



2. ábra. Twain: The Adventures of Tom Sawyer. A bal oldali ábra az eredeti mű alapján készült és a megfelelő szótípusok számát mutatja 100 szó hosszúságú blokkok esetén. A jobb oldali ábrán az eredeti mű típusainak gyakorisága alapján készült modell látható. A modell nem követi azokat a változásokat, amelyek a történethez szervesen nem kapcsolódnak

A rövid, egyenlő hosszúságú intervallumok alkalmasnak bizonyultak arra, hogy a szöveg apró változásainak a jelei is megfigyelhetőek legyenek a grafikonon. Ilyen jellegű változások következhetnek be, amikor például egy új szereplő, helyszín, esemény hosszas bemutatása szakítja meg a történet folyamát, egy olyan új szereplőt beszéltet az író, akinek a stílusa nagyban eltér a korábban megszólaltatott szereplőkétől, idegen kifejezések, mondatok keverednek az egynyelvű szövegbe.
A program által generált modell vissza tudta adni azokat az apró változásokat, amelyek a történet logikus következményei voltak. Ezzel szemben azok az események, amelyek nem illettek a szövegbe, vagy nem tartoztak szervesen a történethez, nem jelentek meg a modellben (3. ábra).
Azt is sikerült megmutatni, hogy a szöveg hosszának fontos szerepe van a típusok megjelenésének szabályszerűségében. Korábbi írások már említették a szövegek hosszának meghatározó jellegét [5]. A megrajzolt függvények alapján állíthatjuk, hogy a típusok megjelenése egyenletesebb novellák esetén, mint regényekben, valamint azt, hogy a bevezetésre kerülő új típusok száma novellák esetén még a szöveg végén is magas (1. ábra). Ezzel magyarázható, hogy hasonló műfajú összefűzött novellák esetén sem találtunk nagy ugrásokat az összefűzési pontoknál még különböző szerzők esetén sem. Az összefűzött novellák viselkedéséből az is látható, hogy a típusok bevezetésének szabályszerűsége nem annyira a szerző, mint inkább a műfaj jellemzője, hasonló eredmények más modellek alkalmazása során is születtek már [6].
Nemcsak irodalmi művek, hanem egynyelvű nyelvkönyvek is feldolgozásra kerültek. A nyelvkönyvek megválasztásának szempontjai már tekintélyes részét képezik a nyelvoktatás módszertani irodalmának [7], de sok-sok szempont között az utolsók között kullognak azok, amelyek a szókészlet tudatos megválasztásáról szólnak. Azt vallják, hogy tanítsanak a nyelvtanárok annyi szót, amennyit csak lehet, de kurzusonként (120-150 óra) legalább 1000 szót, és amennyiben ez lehetséges, a leggyakoribb és „leghasznosabb" szavakat. Az általunk választott nyelvkönyvsorozat elemzése azt mutatta, hogy egy nyelvkönyv szótípusainak megjelenése nem tér el

lényegesen az összefűzött novelláknál megfigyeltektől (1. táblázat). Másik, figyelemre méltó eredmény pedig, hogy rendkívül magas azoknak a típusoknak a száma, amelyek csak egyszer szerepelnek a nyelvkönyvben (2. táblázat).



3. ábra. Twain: The Adventures of Tom Sawyer. A bal oldali ábra pontokkal az eredeti könyv típusainak megjelenését mutatja. Folyamatos vonallal egy 5-pontos simítás eredményét jelöltük. A jobb oldali ábra a modell típusait és a simítás eredményeit szemlélteti. A simított függvényen jól láthatóak azok a szövegrészek, amelyek kiugranak a történet logikus menetéből

1. táblázat. Összefűzött novellák és egy nyelvkönyv szókészletére vonatkozó adatok

Szerző, cím	Blokkok	Típusok	Hapax legomena
Kipling: The Jungle Books	516	4688	2067
Soars: Headway Intermediate	500	4803	2072

2. táblázat. A New Headway nyelvkönyvsorozat köteteinek összehasonlító elemzése

New Headway	Blokkok	Típusok	Hapax legomena
Beginner	163	1539	501
Beginner→Elementary	402	2943	962
Beginner→Pre-Intermediate	719	4550	1628
Beginner→Intermediate	1220	6760	2607
Beginner→Upper-Intermediate	1731	8989	3458

A program magyar nyelvű szövegek elemzését is elvégzi. Angol nyelvű szövegekhez hasonlóan, a típusok megjelenésének szabályszerűsége nem alkalmas arra, hogy azt

szerző azonosításhoz használjuk. Az 1. ábra grafikonjai azonban egyértelműen szemléltetik, hogy a magyar szövegekben a típusok száma magasabb, mint angol szövegekben. Ez azonban nem feltétlenül a magasabb számú szókészlettel magyarázható, hanem a magyar nyelv szóképzési és ragozási szabályaiból következik. Ahhoz, hogy a két nyelv szókészletét, az eredeti és a lefordított mű fordításánál felhasznált szókészletét érdemben össze tudjuk hasonlítani, morfológiai elemzőre és egyértelműsítő programokra lenne szükség.

Összegzés

Az ismertetett modellt és az eredeti szöveget összehasonlítva megtalálhatók a szövegnek azon pontjai, amelyek nem következnek logikusan az előzményekből. A program segítségével azt is szemléltetni tudjuk, hogy egy egynyelvű nyelvkönyv típusai mennyiben térnek el, vagy mennyiben hasonlítanak egy regényhez.
Azt találtuk, hogy az új típusok megjelenésének szabályszerűsége sokkal inkább a szöveg hosszától, a szöveg műfajától függ, mintsem a szerzőtől. A rövid, egyenlő hosszúságú intervallumokra bontott szövegek sem adnak vissza a szerzőről olyan információt, amely alapján azonosítani lehet a szerzőt. További összehasonlítható eredményeket kaphatunk, ha olyan sorozatok elemzését végezzük el, amelyek kötetei bizonyítottan különböző szerzőktől származnak. Vizsgálataink során szeretnénk összehasonlítani korábbi századokból származó, valamint XX. századi műveket, annak érdekében, hogy megvizsgáljuk, találunk-e kimutatható eltérést a típusok megjelenésének szabályszerűségében az idők folyamán. Terveink között szerepel még nyelvkönyvek további feldolgozása is. A nyelvkönyvek megválasztásánál az elsődleges szempont a kiadó lesz. Ennek megfelelően olyan további sorozatokat szeretnénk elemezni, amelyek ugyanattól a kiadótól származnak, illetve olyanokat, amelyek más kiadó művei. A nyelvkönyvek elemzése a most ismertetett módszerrel további szempontokat adhat a nyelvkönyvek szóanyagának megválasztásához.

Irodalomjegyzék

1. Yule G. U.: The Statistical Study of Literary Vocabulary. Cambridge University Press (1944)
2. Baayen R. H.: The Randomness Assumption in Word Frequency Statistics, Research in Humanities Computing 5 Selected Papers from the ACH/ALLC Conference, University of California, Santa Barbara, August 1995 (1996) 17-31.
3. Baayen R. H.: The Effect of Lexical Specialization on the Growth Curve of the Vocabulary. Computational Linguistics 22. (1996) 455-480.
4. Hoover D. L.: Another Perspective on Vocabulary Richness. Computers and the Humanities 37 (2003) 151-178.
5. Baayen R. H.: Statistical Models for Word Frequency Distributions: A Linguistic Evaluation. Computers and the Humanities 26. (1993) 347-363.
6. Baayen H., Halteren H., Tweedie F.: Outside the Cave of Shadows: Using Syntactic Annotation to Enhance Authorship Attribution. Literary and Linguistic Computing, Vol. 11, No. 3. (1996) 121-131
7. Cunningsworth A.: Choosing your Coursebook. Heinemann (1995)

Word Frequency Distribution in English and Hungarian texts

Mária Csernoch[1], László Hunyadi[2]

[1]Institute for English and American Studies, [2]Department of General and Applied Linguistics, University of Debrecen, 4010 Debrecen, Hungary
mcsernoch@hotmail.com

Models based on the frequency of word-types assume that tokens in a text occur randomly. Even though, many different strategies can be followed in the process of selecting words randomly. The best models proved to be those that assume that word-types are binomially distributed. Based on this binomial distribution a model was created first by Baayen [1] than by Hoover [2]. In these models a constant was calculated for each type which occurred m times in the original text. This constant served as the predicted number of types occurring in the text. Summing up these constants for each type a predicted number of word-types can be calculated. However, the model created this way is static, thus it always provides a constant for a selected M ($M \leq N$, where N is the number of tokens in the text). In the era of the new generation of personal computers dynamic models can also be built based on the same assumptions.

The ultimate goal of our studies was to build such a dynamic model. The question was whether this new model can help us to explain the regularities of the introduction of word-types in a text. Furthermore, can we see what the sources are which force the authors to use new types in their works, and can we find any significant sign to use this method to recognize the author of a selected work?

Hungarian and English literary works and English textbooks were analyzed the find regularities in the introduction of word-types in these works. To carry out the study the texts were divided into short, constant-length intervals with a usual length of 100 words. One of the advantages of this method was that the short intervals allowed us to follow minor changes in the texts. Based on the frequency of the word-types in the original text a model, an artificial text was created. Comparing the original and the artificial text we were able to find intervals in the original text which made a kind of stand out. These jumps were found to be responsible for something unpredicted, sometimes illogical events in the discourse of the text. Analyzing the textbooks, we learned that the introduction of word-types in these books showed resemblance to randomly chosen and then concatenated short stories. It seemed that the authors of the textbooks ignored that not only the number of word-types should be increased, but the words should be repeated a certain times in these books.

References

1. Baayen R. H.: The Effect of Lexical Specialization on the Growth Curve of the Vocabulary. Computational Linguistics 22. (1996) 455-480.
2. Hoover D. L.: Another Perspective on Vocabulary Richness. Computers and the Humanities 37 (2003) 151-178.

A szóról és a szófajokról (a számítógépes nyelvfeldolgozás kapcsán)

Bibok Károly

Szegedi Tudományegyetem, Orosz Filológiai Tanszék
6722 Szeged, Egyetem u. 2.
kbibok@lit.u-szeged.hu

Jelen cikk a számítógépes nyelvfeldolgozásnak három, a morfoszintaktikai elemzéssel kapcsolatos kérdéskörét vizsgálja. Ezek a következők: 1. a szövegszavakra bontás (tokenizálás) nehéz esetei, 2. a szövegszavak és a morfoszintaktikai elemzés bemenetének viszonya, valamint 3. a hagyományos szófajok közé be nem illeszthető szavak osztályainak létrehozása.

1 Bevezetés

A Szegedi Tudományegyetem Informatikai Tanszékcsoportja és a MorphoLogic Kft. 2000–2002-ben elkészítette a Szeged Korpusz első változatát (http://www.inf.u-szeged.hu/lll), amely különböző nyelvhasználati területekről összesen 1 millió szövegszónyi, morfoszintaktikailag elemzett (MSD szerint kódolt) és egyértelműsített magyar nyelvű szövegeket tartalmaz [1]. A korpusz előállítási munkálatai közben egy sor olyan szövegfeldolgozási problémával találkoztunk, amelyeknek teljes körű megoldását akkor nem vállalhattuk fel. Ezek közül fogok jelen cikkemben hármat tüzetesen megvizsgálni.

2 Mi a szó?

A nyelvészetben a szó meghatározásával kapcsolatban lehetséges az az álláspont, hogy nem a szót általában, hanem részfogalmait definiáljuk. A nyelvi részrendszereknek megfelelően beszélhetünk fonológiai, morfológiai, szintaktikai és lexikai szóról [2: 76–79]. Egy másik szempontot követve, a szövegszót, a morfoszintaktikai szót és a lexémát különíthetjük el [6: 190–191]. Itt most részletesen nem hasonlíthatom össze a terminus technicusok fenti két halmazát, de a következőket szükséges megjegyezni. A *lexéma* helyett használható a *lexikai szó*. A *morfoszintaktikai szó*-ban nem válik külön a morfológiai és a szintaktikai jelleg. A *szövegszó*-nak nincs megfelelője az első felsorolásban, mert a szövegszó nem a nyelv (langue), hanem a beszéd (parole) egysége. Mivel azonban a számítógépes

nyelvfeldolgozás szövegekkel operál, számunkra mégis ez képezi a kiindulópontot, mégpedig úgy, ahogy a nyelv írott formájában, az írásbeli szövegekben megvalósul.[1]

Kezdjük Papp Ferenc következő meghatározásával: a szövegszó az írás szemszögéből a szövegnek két szóköz határolta darabja [7: 76]. Ha el is tekintünk attól a partikuláris problémától, hogy a szöveg kezdő és végső eleme nem szóközök között áll, marad egy sokkal általánosabb nehézség, amelyre maga Papp Ferenc is utal. Ha egy számítógép az előbbi definíció alapján végezné a szöveg szegmentálását, akkor bizonyos szövegszavak elején és végén írásjeleket, pl. idézőjelet, vesszőt, felkiáltójelet, találnánk, mert ezek szóköz nélkül kapcsolódnak az első, ill. az utolsó betűhöz [7: 77]. Ezért teljesebb Papp Ferencnek az a meghatározása, amely szerint a szövegszó a szövegnek az a részlete, amely két szóköz között helyezkedik el, leszámítva az írásjeleket [6: 190]. Sokszor azonban még ez a meghatározás sem vezet a kellő eredményre és módosításra szorul a szövegszavak két csoportja miatt.

Az első csoportba azok a szövegszavak tartoznak, amelyeknek **a szélein le nem választható/választandó írásjel(ek)** van(nak). Mielőtt rátérnék ezekre a szövegszavakra, megjegyzem, hogy az ilyenek feltételezése egyáltalán nem ellentétes a definíció magját képező „a szövegnek az a részlete" kitétellel, vagyis nincs kikötve, hogy a szövegszavaknak csak betűkből kell állniuk. Tehát az első csoportba sorolhatók a következő esetek. Először, bizonyos rövidítések végére és az arab vagy római számmal írt sorszámnevek[2] után pontot teszünk. Másodszor, az olyan esetben, mint pl. *írásjel(ek)*, a záró zárójel leválasztása hibás elemzést adna (vö. még: *„Hét évszázad magyar versei"-ben*). Harmadszor, a *majd'; bel- és külkereskedelem; gépgyártó, -szerelő és -javító üzem;* stb. kifejezésekben az írásjelek elhagyása megtévesztő lenne. Negyedszer, az egyéb, nem a központozást, tagolást szolgáló írásjeleket sem kell levágni (pl. *a), 10%, 3°, °C, +2, –2*).

A másik csoportot, amely miatt pontosítani kell a fenti definíciót, a **belsejükben írásjelet tartalmazó szövegszavak** alkotják. Ahogy – a fentiekkel összhangban – arab vagy római számmal írt számok szerepelhetnek szövegszóban, ill. szövegszóként, és lehetnek a szövegszavak elején és végén írásjelek, úgy a definíció nem zárja ki azt sem, hogy a szövegszó belsejében írásjel legyen.[3] A kérdés az: Vonatkoztassuk-e ezekre az írásjelekre is a definíció azon részletét, hogy „leszámítva az írásjeleket"? A számítógépes korszak előtt született szövegek szavainak belsejében leggyakrabban a kötőjel fordul elő. A kötőjelnek, mint ismeretes, több funkciója lehet: használhatjuk összetett szavakban, az *-e* kérdőszós szavakban, toldalékok kapcsolására és a szavak sorvégi elválasztására is. Az utóbbi esetben – a központozási szerepet játszó írásjelekhez hasonlóan – nem tekinthető a szövegszó részének. Nyilvánvalóan az nem oldja meg a problémát, ha minden sorvégi kötőjelet eltüntetünk a szövegünkből, hacsak- nem előzőleg különböző gépi kötőjeleket alkalmazva rögzítettük a szöveget (a kettőzött kétjegyű betűkkel még ekkor is vigyázni kell). De mi a helyzet a számítógépes korszak szövegeiben előforduló szóbelseji írásjelekkel (l. pl. URL- és e-mail-címek)? Úgy gondolom, ugyanúgy kell

[1] A szakirodalomban a *szövegszó*-n kívül találkozhatunk a *szóelőfordulás* terminussal [5: 26], illetve a nyelv írott formájában előforduló szövegszót hívják ortográfiai szónak is [8: 1058].

[2] A szöveg részeként szereplő szám – a definíció szerint – szintén szövegszónak tekintendő.

[3] Azt a megszorítást azonban megtehetjük, hogy a szövegnek két szóköz közötti azon részlete, amely csak írásjelet (nevezetesen egy gondolatjelet) foglal magában, nem számít szövegszónak.

eljárni, mint a nem elválasztást szolgáló kötőjel esetében. Az ilyen írásjeleket nem kell kiiktatni a szövegszavakból. Ez azt is jelenti, hogy azt sem tartom elfogadható megoldásnak, ha a (szóköz nélküli) írásjelek mentén szétdarabolnánk őket. (Még szóköz nélküli pont esetén sem kell ezt tennünk.) Ugyanis nem kapnánk ezáltal a morfoszintaktikai elemzés számára értelmezhetőbb kifejezéseket. Ha figyelembe vesszük, hogy az újabb keletű, szóbelseji írásjeleket tartalmazó szövegszavak is elláthatók névszói ragokkal, akkor a sporthírekben olvasható olyan időeredményekre, mint pl. *1:20:36.7, 4:01,95* szintén igaz az a fenti kijelentés, hogy a szétdarabolás nem ad jobb elemzést.

3 A morfoszintaktikai elemzés bemenete

Prószéky Gábor szerint „az ortográfiai szavak definíciója egyértelmű definíciót ad a számítógépes morfológia bemenetére" [8: 1058]. Én azonban úgy vélem, hogy a szöveg szegmentálása útján kapott szövegszavak nem minden esetben felelnek meg a morfológiai elemzés, ill. a morfoszintaktikai kategóriák szerinti elemzés számára. Vannak olyan **szövegszavak, amelyeket szét kell bontani,** és vannak olyanok, **amelyeket egybe kell vonni a morfoszintaktikai elemzés előtt.** Másként megfogalmazva, egy szövegszó magában foglalhat két morfoszintaktikai szót, vagy több szövegszó tesz ki egy morfoszintaktikai szót.

Az első esetre két példát hozok:

1. Az *-e* kérdőszós szövegszavakban előforduló *-e* kérdőszót csak akkor tudjuk külön kategóriaként kódolni (vö. az ÉKsz.[2]-ben: határozószó (simuló kérdőszócska) [9]), ha az ilyen szövegszavakat egy újabb szegmentálás során kettévágjuk.
2. A sporthírekben gyakori kifejezés az olyan, mint pl. *Dortmund (német)-Barcelona (spanyol),*[4] amelyből a *(német)-Barcelona* a 2.-ben mondattak alapján egy szövegszó. Mivel ehhez nem tudunk „értelmes" kódolást rendelni, itt is először újabb szegmentálásra van szükség: három részre osztjuk, ugyanis a kötőjel is külön elem.

A második esetre jóval több példát tudok mutatni.

1. A több szövegszóból álló tulajdonnevek: pl. *Szegedi Tudományegyetem, New York-i.* Az utóbbi különösen jól szemlélteti a többtagú tulajdonnevek egyes tagjainak egybetartozását, hiszen a *York*-ból képzett *-i* képzős alak a *yorki.*[5]
2. Többtagú rövidítések: *i. sz., i. e., i. m.* stb.
3. A többtagú számnevek:

a) Hátulról számított hármas számcsoportok szerint szóközzel tagolt (arab) számok. Gondoljuk meg, milyen eredményre vezetne, ha a *3 000 000*-t mint három szövegszót elemeznénk morfoszintaktikailag. Ahhoz, hogy a *000*-ról értelmes dolgot mondhassunk, fel kell tételeznünk egy morfoszintaktikai osztályt a számok részeit képező számok számára. De ennek alapjául csak az az írásgyakorlat szolgálna, amely szerint a számmal írt számneveket hármas számcsoportokba tagoljuk.

---

[4] Szövegeinkben sajnos nincs különbség kiskötőjel és nagykötőjel között.

[5] Ezen az alapon a *New York-i,* ha nem is egy szövegszó, egy ortográfiai szónak tekinthető.

b) Az (arab) számokat és az *ezer, millió* stb. szavakat tartalmazó számnevek. Ha ezeket nem (külön írt) összetételeknek,[6] hanem jelzős szerkezeteknek tartanánk, akkor a csak betűvel, valamint a csak számmal írt számnevektől eltérően kezelnénk őket.

4. Kötőjeles kifejezés egyik eleme többtagú, pl. *Ferencváros-Vác FC.*
5. A vagylagosságot kifejező „/" jellel összekapcsolt kifejezés egyik eleme többtagú, pl. *főnevek/főnévi csoportok.*
6. Elmaradó közös tagú kifejezések, pl. *bel- és külkereskedelem; gépgyártó, -szerelő és -javító; Tömörkény- és Gárdonyi-szerű,* kivéve *Tömörkény- és később Gárdonyi-szerű.* Ezekben az esetekben tulajdonképpen a morfológiai és szintaktikai tulajdonságok ütközéséről van szó. A *belkereskedelem* mint összetétel egy morfológiai szó, de szintaktikailag nem egységként viselkedik: az elő- és az utótag elválhat egymástól a mellérendelés során. Ez a kettős jelleg lehetővé teszi, hogy kétféleképpen adjunk számot a *bel- és külkereskedelem*-ről. Vagy besoroljuk a morfoszintaktikai kategorizálás révén a főnevek közé, nem törődve a mellérendeléssel, amely – az alárendelő szerkezetekkel szemben – a szintaxisnak is másodlagos vizsgálati tárgya. Vagy szintaktikai szerkezetként kezeljük és feltételezünk egy, a szélein kötőjelet tartalmazó morfológiai osztályt. Erre az utóbbi megoldásra is van lehetőség a 4.-ben vázolandó klasszifikációs rendszer keretében. Ugyanis a valószínűleg nagyon ritkán előforduló *Tömörkény- és később Gárdonyi-szerű* és az ehhez hasonló kivételek miatt mindenképpen szükséges lesz bevezetni az elöl és/vagy a végükön kötőjelet tartalmazó szavak morfológiai osztályát (Hatvani Csaba, személyes közlés).

4 Hány szófaj van?

Itt nem a szófajok definíciós kritériumairól és a szófajok elhatárolásáról folytatott vitáról kívánok szólni. Hadd kezdjem annak leszögezésével, ami már a 3.-ból is kitűnhetett. A morfoszintaktikai, ill. a morfológiai és a szintaktikai szó mögött a lexikai szó (lexéma) és a termékeny módon alkotott létező vagy potenciális szó húzódik meg (esetleg még számolhatunk a mellérendeléssel is).[7] Ezek pedig morfoszintaktikai osztályokba sorolhatók. Ugyanakkor az is nyilvánvaló a fentiekből, hogy ha egy korpusz különböző nyelvhasználati rétegekből tevődik össze, akkor a **hagyományosan szófajokon túl** és az eddig alkalmazott MSD-kódrendszert kibővítve **újabb osztályokat kell létrehozni,** hogy a klasszifikációt maradéktalanul elvégezhessük (vö. számítógépes szaknyelvbeli URL- és e-mail-címek vagy a sporthírekbeli számok és írásjelek kombinációjából álló meccs- és időeredmények). Mielőtt a nyolc javasolt osztályt és alosztályaikat bemutatnám, előrebocsátok három megjegyzést. Először, még ha – ugyanúgy, mint a hagyományos szófajok esetében – formális meghatározásra törekszünk is, nem kerülhető el a jelentésre való hivatkozás

[6] Abban, hogy ezeket összetételeknek minősítjük-e, nincs jelentősége a különírásnak. Vö.: ha a szám helyett is betűt használunk, akkor egybeírjuk az ilyen számnevet.

[7] A létező és a potenciális szavak közötti különbségtételről l. [3: 148].

sem.[8] Másodszor, az új „szófajok" – az önállóan használt toldalékok osztályának kivételével (pl. a *-ság* a *-ság képzős főnevek* kifejezésben) – nyitott osztályok, azaz elemei korlátlanul szaporíthatók, például a szóalkotáshoz hasonló módokon.[9] Harmadszor, mivel eddig az osztályok megnevezéseinél a mnemonika és a magyarul nem beszélő felhasználók szempontjai részesültek előnyben, csak körülíróan és példákkal szemléltetve tudom bemutatni a **javasolt osztályokat és alosztályaikat.**

1. Elektronikus:
a) e-mail-címek, pl. *A bubo@doktor.hu címről...,*
b) webhelyek, pl. *A www.huninet.hu-ról...,*
c) számítógépes útvonal, pl. *A C:\CONFIG.SYS nevű...,*
d) fájlkiterjesztés, pl. *A .DOC és a .RTF fájlok...,*
e) egyéb.
2. Számok:
a) (sport)eredmények (meccs-, időeredmények), pl.: *1:20:36.7, 4:01,95, 2:0, 2-0,*
b) (csak) előjelet tartalmazó (egész és nem egész) számok, nem tartozik ide a telefonszámban az ország hívószáma az előtte álló „+" jellel,
c) időpont-megjelölések, dátumok, amelyek szóköz nélküli pontot, kettőspontot, kötőjelet vagy „/" jelet tartalmaznak, pl. *10.35, 10:35* (= *10.35*), *2003-01-06, 2003.01.06., 01/06,* de nem tartozik ide: *1984/85-ben, 1984/1985-ben, 1984/85. (tanév), 1984/1985. (tanév),*
d) (csak) pontot tartalmazó szám, pl. *139.000* (= *százharminckilencezer*), *80.5 MB* (= *80,5 MB*),
e) százalékjelet („%") tartalmazó kifejezések, pl. *(+)10%, 40.2%,*
f) fokjelet („°") tartalmazó kifejezések, pl. *(+)3°,* de nem: *(+)3 °C,*
g) arány, pl. *1653kJ/1000g, 1653 kJ/100 g, 1:10,*
h) méret, pl. *1024x768,* ide tartozik még: *2x* (= *2-szer*),
i) képletek, aritmetikai kifejezések, pl. *2rπ, 2+2 =4, 2 + 2 = 4* (összevonás után),
j) egyéb.
3. Indexként bármilyen karaktert tartalmazó kifejezés:
a) alsó index,
b) felső index, pl. *dpi**-vel, quattro®,* kivéve: m^2, cm^3, $3°$.
4. Különféle azonosítók, pl. szabvány jelzete, igazolványszám, iktatószám, ügyiratszám, alvázszám, motorszám, rendszám *(ABC-123, ABC123),* ISBN, könyvtári jelzet, *1/a, 1/A, III/1* típusú jelzet (a *3/4, 10/100, 10/1974* törtszámnév is lehet), írásmű részének jelzése (*1.1.2., 1.1.2, a)*), utak, géptípusok jelzése *(E5, M0, MiG-27, T-34, TU-154,* tulajdonnévként kódoljuk, ha a betű- és/vagy számjelzés szóhoz tartozik, pl. *Apollo-11, Boeing XXX, Commodore 64),* telefonszám (*473-1470,* de ha évszám, akkor „sima" számnév), irányítószám („sima" számnév is lehet).
5. Szónál kisebb tokenek, pl. *A -ság képzős főnevek..., A -tól/-től ragos eset...,* de nem ide tartozik: ***bel- és külkereskedelem*** stb.

---

[8] A főnév és az ige formális alapú definiálására l. [4: 148–149, 209–210]. Ugyanakkor a melléknévnek a főnévtől és az igétől való szófaji elkülönítéséhez mindenekelőtt szemantikai kritériumokat kell keresni [4: 181].

[9] A nyitott és a zárt osztályokról a hagyományos szófajok kapcsán l. [2: 95].

6. Kötőjellel kezdődő vagy végződő tokenek az olyan kifejezésekben, mint pl. *Tömörkény- és később Gárdonyi-szerű.*
7. Nem magyar, vagyis idegen nyelvű szavak, ill. kifejezések.
8. Rossz helyesírással írt magyar szavak, amelyek nem homonimak más magyar szavakkal, pl.: *éccaka,* de nem tartozik ide: *aszt* stb.

Fontos kiemelni, hogy ugyanúgy, ahogy a hagyományos szófajoknál, itt is számtalan esetben találkozhatunk **ambiguitási problémával,** azaz azzal, hogy egy egység – a kontextus ismerete nélkül – különböző osztályokba, alosztályokba sorolható be. Ebből a szempontból l. még egyszer a különféle azonosítóknál említett példákat.

5 Összegzés helyett

A *Mondatszintaxis gépi tanulása (gépi tanulási módszerek a magyar nyelv szintaktikai szabályainak létrehozására)* c. IKTA-pályázat keretében (№ 37/2002, vezetője: Gyimóthy Tibor) az SZTE Informatikai Tanszékcsoportjában most folyik annak a számítógépes programnak a tesztelése, amelynek segítségével a fentebb vázolt szegmentálási és klasszifikációs problémák automatikusan kezelhetőkké válnak. Továbbá: Alexin Zoltán és Hatvani Csaba közreműködésével elkészítettem azt az útmutatót, amelynek alapján egyetemi hallgatók ellenőrizni tudják a Szegedi Korpusz e programmal történő elemzésének eredményeit és végrehajthatják a manuálisan elvégzendő feladatokat (szövegszavak összevonása és egyértelműsítés) is.

Végezetül megemlítem, hogy a jelen cikkben kifejtettek nemcsak a nyelvfeldolgozás területén, hanem helyesírás-ellenőrző programok tökéletesítésében és szótárak (pl. gyakorisági szótár) készítésében is hasznosíthatók.

Irodalom

1. Alexin, Z., Csirik, J., Gyimóthy, T., Bibok, K., Hatvani, C., Prószéky, G., Tihanyi, L.: Manually Annotated Hungarian Corpus. In: Proceedings of EACL. Budapest (2003) 53–56
2. Kenesei I.: Szavak, szófajok, toldalékok. In: Kiefer F. (szerk.): Strukturális magyar nyelvtan I. Morfológia. Akadémiai Kiadó, Budapest (2000) 75–136
3. Kiefer F.: A szóképzés. In: Kiefer F. (szerk.): Strukturális magyar nyelvtan I. Morfológia. Akadémiai Kiadó, Budapest (2000) 137–164
4. Kiefer F.: Jelentéselmélet. Corvina, Budapest (2000)
5. Lengyel K.: A nyelvi egységek szinteződése. In: Keszler B. (szerk.): Magyar grammatika. Nemzeti Tankönyvkiadó, Budapest (2000) 24–33
6. Papp, F.: Morfologija. In: Papp, F. (szerk.): Kurs sovremennogo russkogo jazyka. Tankönyvkiadó, Budapest (1968) 189–423
7. Papp F.: Szövegszó, szóalak, lexéma. Magyar Nyelvőr 98 (1974) 76–82
8. Prószéky G.: A magyar morfológia számítógépes kezelése. In: Kiefer F. (szerk.): Strukturális magyar nyelvtan I. Morfológia. Akadémiai Kiadó, Budapest (2000) 1021–1063
9. Pusztai F. (főszerk.): Magyar értelmező kéziszótár. Második, átdolgozott kiadás. Akadémiai Kiadó, Budapest (2003)

More about Words and Parts of Speech (Concerning Natural Language Processing)

Károly Bibok

University of Szeged, Dept. of Russian Philology
Egyetem u. 2. 6722 Szeged, Hungary
kbibok@lit.u-szeged.hu

In 2000–2002 a consortium of the University of Szeged and MorphoLogic Ltd. (Budapest) developed a morpho-syntactically parsed and disambiguated corpus for Hungarian up to one million text words (tokens) without punctuation characters (for Version 1.0 see http://inf.u-szeged.hu/lll). During this project some problems of the natural language processing (NLP) were realized, the complete solution of which could not been attempted. The following can be mentioned: eliminating mistakes made by text segmentation program, treating text words as morpho-syntactic units, and creating word classes for tokens which do not fit into the traditional parts of speech. The current paper deals with these three problematic cases which more attention should be paid to in the future.

For the written form of a language, a text word is a minimal fragment of a text occurring between spaces, not including punctuation marks (Papp 1968: 190). A program that carries out text segmentation task can be based on this classical definition of the text word if it is made more precise with respect to: 1. tokens containing not detachable punctuation marks on one or another side (e.g. abbreviations ending in points) and 2. tokens having punctuation marks in their internal parts (e.g. e-mail addresses).

Some sequences of text words, however, have to be considered grammatical forms of units, lexical or generated by rules of word-formation and coordination. Therefore, the components, i.e. text words, of such units (e.g. of a complex proper noun) should be drawn together before the morpho-syntactic parsing starts.

Furthermore, in case of a very large and stylistically heterogeneous corpus like ours, the traditional set of parts of speech should be extended in order to classify tokens containing special (punctuation) marks ("://", "\", "@" etc.) in the sub-language of computing or tokens consisting of combinations of numbers and punctuation marks in sports news.

Magyar szövegek természetes nyelvi előfeldolgozása

Miháczi András, Németh László, Rácz Miklós

Mihaczi.Andras.Janos@stud.u-szeged.hu
NemethL@gyorsposta.hu
Racz.Miklos@stud.u-szeged.hu
Szegedi Tudományegyetem, Informatikai Tanszékcsoport

Kivonat. A természetes nyelvi szövegek előfeldolgozásának feladata a szöveg mondatokra, szavakra bontása, tokenizálása (tokennek nevezzük a legkisebb önálló jelentéssel bíró szövegegységet). Ehhez szorosan kapcsolódik az úgy nevezett nyílt tokenosztályokba tartozó egyes tokenek felismerése. Ezek olyan tokenek, amelyekben speciális (írás)jelek vagy szóközök vannak. Az előfeldolgozás része a tulajdonnevek felismerése is, hiszen itt nagyméretű, tulajdonneveket tartalmazó, szótárakat kell használni A feladatok megoldására kipróbáltunk reguláris kifejezések alapján generált automatát, valamint döntésifa-tanuló algoritmusok által tanult szabályokat.

1. Bevezetés

A természetes nyelvi szövegek feldolgozásának kiindulópontja egy egyszerű, formázatlan szöveg. Az előfeldolgozás feladata a szöveg mondatokra, szavakra bontása, tokenizálása.

A szöveg mondatokra bontása az esetek nagy részében egyszerű feladat, a probléma a lehetséges mondatzáró írásjelek adott környezetben való értelmezése, ugyanis vannak olyan esetek, ahol nem mondathatárt jelölnek.

A szóhatárok a legtöbb esetben egyértelműek, hiszen a szavakat szóközök választják el. A feladat itt is a különleges esetek kezelése, előfordul, hogy az írásjelek hozzátartoznak egy szóhoz. Ehhez szorosan kapcsolódik az úgy nevezett nyílt tokenosztályokba tartozó egyes tokenek felismerése.

A tulajdonnevek felismerése azért az előfeldolgozás része, mert befolyásolja a szegmentálást, ugyanis egy tulajdonnév több szóból is állhat. Ez a folyamat általában szótárak használatával történik, illetve meghatározhatók bizonyos szabályok, melyek alapján összeállnak a tulajdonnevek.

A feladatok megoldására több módszert is kipróbáltunk. A munkában nagy segítséget jelentett az annotált Szeged Korpusz[1]. A korpusz alapján a mondat- és szószegmentálásra egyaránt kipróbáltunk döntésifa-tanuló algoritmusokat. A tulajdonnevek felismerése nagy méretű szótárak használatával történt. A

[1] Ezúton szeretnénk köszönetet nyilvánítani a Szeged Korpusz készítőinek, az SZTE Informatikai Tanszékcsoport, a MorphoLogic Kft. és az MTA Nyelvtudományi Intézet munkatársainak szakmai segítségükért, valamint az általuk gyűjtött adatbázisokért.

tulajdonnévszótár összeállításában is segítségünkre volt a korpusz. Ezen kívül felhasználtunk célszótárakat. Az eredmények tesztelése ugyancsak a Szeged Korpusz alapján történt [2].

2. Szegmentálás

2.1. Mondat és szószegmentálás

A szövegfeldolgozás első lépése a szöveg mondatokra és szavakra bontása. A szavak és írásjelek jelentik a szöveg alapegységeit, ennél részletesebb felbontással nem foglalkozunk. A feldolgozás során ezekhez rendelünk attribútumokat (például szófajok), vagy vonjuk őket össze nagyobb csoportokba (például főnévi szerkezetek). A szavakon itt nem csak a hétköznapi értelemben vett szavakat értjük, ide tartoznak a nyílt tokenosztályok elemei, illetve a több tagból álló tulajdonnevek is.

A mondatokra bontás szintén fontos, ez határozza meg az összetartozó szavakat. Ez főleg a szöveg szemantikai feldolgozásánál számít, de jelenthet információt például a szófaji egyértelműsítés számára is.

Az is fontos, hogy a két fázis milyen sorrendben követi egymást, hiszen fel lehet használni az előző rész eredményeit. (A mondatokat szavak alapján bonthatjuk, illetve a tokenek meghatározásánál segíthet, hogy nem léphetjük át a mondathatárt.)

2.2. Nyílt tokenosztályok

Egy úgynevezett nyílt tokenosztály elemeit valamilyen meghatározott közös szintaktikai tulajdonság megléte sorolja egy osztályba. Ezek az osztályok formális nyelvtannal definiálható potenciálisan végtelen elemszámú halmazok, azaz a szabályok ismételt alkalmazásával – elvileg – végtelen sok tokent állíthatunk elő. Ilyen lehet a személynevek, az időtartamot jelentő kifejezések osztálya, a webcímek vagy az azonosítók. Az ilyen szavak két csoportba oszthatók. Vannak, amelyek a már meglévő MSD kódokkal jelölhetők, a többi kódolására új MSD kódok kerültek bevezetésre. Ezek rendszerezését Dr. Bibok Károly végezte el.

2.3. Tulajdonnevek

A tulajdonnevek halmaza részhalmaza a főnevek halmazának. Automatikus felismerésük azért nehéz, mert a szövegben előfordulhat akár a világ bármely tájáról vett tulajdonnév is. Ezekre adhatunk bizonyos egyszerű szabályokat, de a döntő szerepet a tulajdonnevek felsorolása jelenti. Ezért a tulajdonnév-felismerésnél döntő szerepet játszik a szótár használata. Nehézséget okoz, hogy a tulajdonnevek száma dinamikusan nő.

3. Szakérői tudást felhasználó feldolgozás

A következő fejezetben olyan módszereket mutatunk be, melyek alapja a szakértői tudás, valamint a szakértők által megfogalmazott szabályok alkalmazása.

3.1. Huntoken

A Huntoken szövegfeldolgozó program a bemeneti szöveget mondatokra és szavakra bontja, illetve a nyílt tokenosztályokba eső szavak egy részét felismeri, és a megfelelő MSD kóddal jelöli. A program bemenete ISO-8859-2 karakterkódolású szöveges állomány, amely a HTML 4 szabvány ISO-8859-1-es karakterentitásait is tartalmazhatja. A program kimenete a Szeged Korpusz készítése során használt XML-alapú formátum.

A Huntoken program csőbe köthető szűrőprogramokból áll, amelyek a GNU Flex lexikaielemző-generátorral készültek. A csővezetéket a huntoken parancs indítja el. A csőbe kötött szűrőprogramok szerepét a következő bekezdések foglalják össze:

A hun_clean szűrő normalizálja a bemenő szöveges állományokat a következő fontosabb műveletek elvégzésével: ismétlődő szóköz értékű (későbbiekben szóköz) karakterek törlése, ismétlődő üres sorok és közbeékelt szóközök törlése, sor eleji és sor végi szóközök törlése, nem törő szóközök szóközzé alakítása, az összes ISO-8859-2-ben szereplő ISO-8859-1-es entitás karakterré alakítása (például: á → á).

A hun_sentence szűrő <s> nyitó- és </s> zárócímke közé zárja a felismert mondatokat, vagyis elvégzi a mondatra bontást. A mondatra bontáshoz viszonylag kevés számú Flex szabály lett megadva (<10), de a szabályok között erősen összetettek is akadnak.

A hun_abbrev program ismert rövidítések, és más beépített szabályok alapján felülbírálja, és szükség esetén módosítja a hun_sentence által megállapított mondathatárokat, valamint a hun_token által megállapított szóhatárokat. Hasonló számú szabályt tartalmaz, mint a hun_sentence szűrő. Összehasonlításul Aberdeen és munkatársai több mint 100 szabályt alkalmaztak mondatra bontó Flex alkalmazásukban [5].

A hun_sentence és hun_abbrev páros a Szeged Korpusz szövegein 1% mondatra bontási hibát követ el. Kifinomultabb módszerek felhasználásával nagyobb mértékű javulás lenne elképzelhető. Angol nyelvű szövegeken, szófaji és egyéb információk felhasználásával a 0,2% hibahatárt sikerült megközelíteni a mondatra bontásban [3].

A hun_token a legnagyobb és a legösszetettebb szűrő. A szóra és írásjelekre bontás mellett a nyitott tokenosztályokba eső szavak jelentős részét felismeri, és a megfelelő MSD kóddal látja el. A következő nyitott tokenek felismerését végzi el: toldalékmorfémák, elektronikus címek, e-mail, webhely, útvonal, fájlkiterjesztés, indexek (trade mark, registered trade mark), számok: (sport)eredmények, előjeles egész számok, időpont, dátum, pontot tartalmazó számok, százalékjelet tartalmazó számok, fokjelet tartalmazó számok, arány (SI mértékegységgel), méret × jellel, képletek, azonosítók (szabvány jelzete, telefonszám, írásmű része, ISBN kód, rendszám, egyéb kötőjellel kezdődő, vagy végződő szavak, számmal és betűvel jelölt számok

A hun_token több mint 200 szabályt tartalmaz, amelyek összehangolt működését csak a beépített tesztrendszerrel lehetett biztosítani a fejlesztés során. Minden egyes új szabály, vagy szabálymódosítás esetén is ellenőrizve lett, hogy a többi szabály tevékenysége nem módosult a változtatások hatására. A Flex nyomkövető üzemmódja pedig lehetőséget biztosított arra, hogy egy-egy bonyolult, nyitott tokenosztályba eső szó felismerését lépésről lépésre tanulmányozni lehessen. A következő példa a 640x480 típusú tokeneket ismeri föl, és az alábbi outputot generálja:

```
[0-9]+("x"|"&times";"?)([0-9]+)?

<w>640x480
<anav>
<msd><lemma>640x480</lemma>
<mscat>[Onm-sn]</mscat></msd>
</anav>
</w>
```

A teszteredmények alapján a Huntoken program szövegfeldolgozó képességei elérik, és pontosságban meg is haladják a kézi szövegfeldolgozását.

3.2. Tulajdonnév felismerés

A tulajdonnevek szövegben való megtalálásához szükséges a felismerő nagyfokú lexikai tudása. Tehát egy ilyen programnak nagy méretű, tulajdonneveket tartalmazó szótárat kell kezelnie.

A mi esetünkben kézenfekvőnek látszott, hogy jelentős mennyiségű tulajdonnevet nyerhetünk az annotált Szeged Korpuszból. Első megközelítésben kigyűjtöttük a korpuszból a tulajdonnevek összes előforduló szóalakját. Mindegyikhez hozzárendeltük a megfelelő szótövet, valamint a szófaját meghatározó MSD kódot. Ez a kiindulási szótár jó alapot jelent a természetes nyelvi szövegekben való tulajdonnév felismeréshez. Az így kapott szótár 18 286 különböző szóalakot tartalmaz. A különböző szótövek száma 14 027 volt. [2]

A tulajdonnevek szótárban való tárolására két lehetséges megoldás mutatkozik. Az egyik szerint a szótárba eltárolásra kerülnének a tulajdonnevek ragozás szempontjából vett összes szóalakja. A magyar nyelv erősen agglutináló jellegű, vagyis a különböző nyelvtani funkciókat legtöbbször toldalékokkal (ragokkal, jelekkel, képzőkkel) fejezi ki, azaz egy adott tulajdonnévnek akár több száz ragozott szóalakja is lehet. Ez nagymértékben megnöveli a szótár méretét, és így a szótárban való keresés is hosszabb időt vesz igénybe. A másik lehetőség szerint a szótárba csak a tulajdonnevek alanyi esetű, azaz ragozatlan alakját vesszük fel. Ekkor a szótárban való keresés a szótár kisebb mérete miatt gyorsabban megvalósítható, viszont ekkor a ragozás problémáját bizonyos toldalékolási szabályok megadásával kell kiküszöbölnünk. Ezen szabályokhoz fel kell venni a főnevekhez kapcsolódó lehetséges toldalékokat (például: -ban/ben, -on/en/ön, -nak/nek). A toldalékok vizsgálatának problémája nagyon összetett, nézzük néhány összetevőjét: A toldalékok szótőhöz való kapcsolásakor figyelembe kell venni például a hangrend szerinti

illeszkedést, a mássalhangzók hasonulását, kötőhangok közbeékelését. Ilyen és ehhez hasonló szabályok sokasága fogalmazható meg a toldalék vizsgálatával kapcsolatban, különösen a nem magyar tulajdonnevek esetén.

A tulajdonnév felismerési fázist a természetes nyelvi szöveg feldolgozási folyamatában a szöveg mondatokra és szavakra bontása után valósítottuk meg. Tehát a feladat az, hogy az egymás után következő szavakról megállapítsuk, azok tulajdonnevek, vagy sem. Beszélhetünk egy-, illetve többtagú tulajdonnevekről. Az egytagúak egy szóból állnak, a többtagúak több, szóközzel elválasztott szóból állnak. Többtagú tulajdonnevek esetén előfordulhat, hogy a szavak között írásjelek is vannak. A tulajdonnevek jelölésénél a többtagúakat összevonjuk, azaz a feldolgozás későbbi fázisai számára azok már egy szóként jelennek meg. A tulajdonnevekhez a felismeréskor szótövet és szófajának, valamint toldalékolásának meghatározásához MSD kódot rendelünk. Egy tulajdonnév TEI XML jelölése:

```
<w>Magyar Hanglemezkiadók Szövetségét
<ana>
<msd><lemma>Magyar Hanglemezkiadók Szövetsége
</lemma><mscat>[Np-sa]</mscat></msd>
</ana>
</w>
```

Tekintve a lehetséges tulajdonnevek nagy számát, nincs lehetőségünk minden tulajdonnevet felvenni a szótárba. Többtagú tulajdonnevek esetén az egyes tagok kombinációjával rengeteg tulajdonnevet ismerhetünk fel. Például egy személynév a legtöbb esetben egy vezetéknévből és egy keresztnévből áll. Az összes személynév eltárolására nincs módunk, de fölvehetjük egy listába a keresztneveket és a vezetékneveket. Így egy szabály alkalmazásával, miszerint egy személynév vezetéknévből és keresztnévből áll, már a legtöbb személynevet föl tudjuk ismerni.

Tulajdonnevek felismerésére alkalmaztunk, reguláris kifejezésekkel megadott szabályokat, valamint célszótárakat. Ilyen célszótárban vannak például a keresztnevek, vezetéknevek, helységnevek, nagyvárosok nevei, cég- és intézményutótagok (kft., rt., Iskola, Tudományegyetem, stb.).

4. Gépi tanulás alkalmazása szegmentálásra

A mondat és szószegmentálásra egyaránt kipróbáltunk döntésifa-tanulást. A tréning és tanulóadatok előállításában fontos szerepe volt a Szeged Korpusznak. Mindkét problémánál az (írás)jelek környezetének vizsgálata a kritikus feladat. A mondat- és szószegmentálásnál is azt a döntést vizsgáltuk, hogy bizonyos karaktereknél illetve tokeneknél van-e mondat-, vagy szóhatár.

A legnehezebb a tréning adatok megfelelő kiválogatása volt. Az esetek túlnyomó többségében nagyon kevés, nagyon egyszerű szabály érvényesül (például mondat végi pont, és utána nagybetű), ezekkel általában el lehet már érni a 90%-os pontosságot. A tanulás során azonban annyira elnyomják a speciális eseteket, hogy azokat az algoritmus zajnak gondolja. Ezen problémákat azzal próbáltuk kivédeni, hogy a

triviális eseteket kihagytuk (például szószegmentálásnál a csupa betűből álló szavakat, ahol nem jelent problémát a szegmentálás).

A csoportosításra két módszert is kipróbáltunk, az egyiknél minden egyes karaktert külön vizsgáltunk, a másiknál próbáltunk őket csoportosítani. Utóbbi esetben betűsorozatokat, számsorozatokat egy egységként kezeltük, figyelembe véve például, hogy a token kezdőbetűje kis vagy nagybetű. Az írásjelek minden esetben darabonként külön csoportot képeztek. Az attribútumok minden esetben a kérdéses és az azt körülvevő tokenek voltak. A környezet mérete nem volt számottevő befolyással az eredményre, a döntési fákban és a belőlük képzett szabályokban csak ritkán fordultak elő a vizsgált, az azt követő, illetve az azt megelőző tokenektől különböző token.

A döntési-fa tanulásra Ross Quinlan C 4.5 algoritmusát használtunk [4], kipróbálva a paraméterek különböző variációit (attribútumok csoportosítása, iterált faépítés, fa vágása, szabályok, stb.).

Az eredmények első megközelítésben jók lettek, a hiba összességében 0,5 és 2% között mozgott. Ez jobb, mint ha csak a triviális szabályokat alkalmaznánk. A nagy döntési fákból csak kevés használható szabály született, ezek száma 7 és 15 között mozgott a tanulási paraméterektől függően. Egy részük triviális (például mondatzáró írásjel és nagybetűs szó esetén van mondathatár), a többi nyelvészetileg nem értelmezhető, a speciális eseteket kezelő szabály.

5. Jövőbeni tervek

A szegmentálással kapcsolatos jövőbeli feladatok között szerepel a tudásalapú rendszerek továbbfejlesztése és a tanuló rendszereknél leírt problémák (tréning adatbázis megfelelő összeválogatása) további vizsgálata.

A tulajdonnév felismerés kapcsán a ragozott alakok pontos felismerése a cél. Ebben segítséget jelent a Szeged Korpuszból kigyűjtött lehetséges toldalékok listája.

6. Irodalomjegyzék

[1] Alexin, Z., Csirik, J., Gyimóthy, T., Bibok K., Hatvani, Cs., Prószéky, G., Tihanyi, L.: Manually Annotated Hungarian Corpus in Proc. of the Research Note Sessions of the 10th Conference of the European Chapter of the Association for Computational Linguistics EACL'03, Budapest, Hungary 15-17 April, pp. 53-56 (2003)

[2] http://www.inf.u-szeged.hu/lll

[3] Mikheev: Periods, Capitalized Words, etc., Computational Linguistics, Vol. 28., 3., 2002, 289 o.

[4] J.R. Quinlan: C4.5: Programs for Machine Learning, Morgan Kaufmann, 1993.

[5] Aberdeen, John S., John D. Burger, David S. Day, Lynette Hirschman, Patricia Robinson, and Marc Vilain. 1995. Mitre: Description of the alembic system used for MUC-6. In Proceedings of the Sixth Message Understanding Conference (MUC-6), Columbia, Maryland, November. Morgan Kaufmann.

Natural language preprocessing on Hungarian language texts

Miháczi András, Németh László, Rácz Miklós

Mihaczi.Andras.Janos@stud.u-szeged.hu
nemethl@gyorsposta.hu
Racz.Miklos@stud.u-szeged.hu

University of Szeged, Department of Informatics

The foundation of natural language processing is plain text. The purpose of preprocessing is to divide the text into digestible units, such as sentences, words, punctuation marks, and other special tokens.

In most cases, the segmentation of sentences is a simple task. The main source of the problem is the interpretation of possible sentence-final punctuation marks, because sometimes it is not the sentence boundary that they represent.

Word boundaries are in most cases unambiguous and easily recognizable since words are separated by whitespaces. However, there are some special cases to be handled, e.g. in the case when the punctuation mark forms an actual part of the word.

The recognition of tokens from open token classes is a closely related problem to the aforementioned ones. These tokens contain special characters (e.g.: comma, dot, hyphen, quotation marks, etc.) or whitespaces.

The recognition of proper names is also a similar problem. Proper name recognition has to be included in the preprocessing phase as well, since it can influence the segmentation of words based on the fact that a proper name may consist of more than one word. The recognition is usually conducted by using dictionaries, and special rules can be defined to build the proper names. (e.g.: person name = first name + last name)

To solve the above described problem, we tried different methods using the Szeged Corpus (an annotated corpus of 1.2 million words developed by the University of Szeged, Department of Informatics and MorphoLogic Ltd). For sentence and word segmentation we have applied decision tree learning algorithms. The aim of learning was to decide whether punctuation marks form part of the words or mark the end of a sentence. The recognition of proper names was aided by large dictionaries containing all proper names from the corpus, and special dictionaries and rules (created and defined by Research Institute for Linguistics at the Hungarian Academy of Sciences) were also used. The results produced by the recognition methods were also tested on the corpus.

In our paper, we present a preprocessing system (forming the initial part of a module chain) developed by ourselves and other existing methods. The preprocessing module recognizes the sentence and word boundaries, special tokens and proper names. The results are passed on to the next module – the morpho-syntactic processing.

Magyar ismeretlenszó-elemző program fejlesztése

Novák Attila[1,2], Nagy Viktor[2] és Oravecz Csaba[2]

[1] MorphoLogic Kft., Budapest
[2] MTA Nyelvtudományi Intézet, Budapest
{novak,nagyv,oravecz}@nytud.hu

Kivonat Nagy korpuszok számítógépes feldolgozása során elkerülhetetlenül beleütközünk abba a problémába, hogy a szövegekben szereplő szóalakok igen jelentős részét nem tudja a rendelkezésre álló morfológiai elemzőprogram elemezni, mert hiányzik az adatbázisából a szó töve. Ugyanakkor ezeknek az elemezhetetlen szóalakoknak a nagy része tartalmaz toldalékokat, ezért valamilyen módon ezeket is elemezni kell. Ennek a problémának a kezelésére olyan hibrid eljárást lehet alkalmazni, amely szimbolikus parciális morfológiai elemzőből és egy olyan statisztikai alapú eszközből áll, amely az első lépésben a szimbolikus ismeretlenszó-elemző által előállított hipotézisteret a kívánt mértékűre szűkíti.
Kulcsszavak: ismeretlenszó-elemzés, morfológiai elemzés, eloszlások hasonlósága, statisztikai egyértelműsítés

1. Bevezető

A magyarhoz hasonló agglutinatív nyelvek számítógépes feldolgozása során a nyelvben előforduló lehetséges szóalakok igen magas száma miatt a morfológiai elemzés alkalmazása gyakorlatilag megkerülhetetlen. Ez a lépés nem helyettesíthető egyszerű szótárból történő lekérdezéssel [1], hiszen egy ilyen szótárnak szinte az összes lehetséges szóalakot tartalmaznia kellene, ez pedig technológiailag kezelhetetlen lenne. Kézenfekvő megoldás egy morfológiai elemző eszközt alkalmazni, amely egy tőtárra támaszkodva képes az inflexiós, a produktív derivációs és szóösszetételi jelenségeket kezelni, valamint adott szóalakokhoz a tövüket hozzárendelni.

Nagy korpuszok számítógépes feldolgozása során viszont kikerülhetetlen az a probléma, hogy a szövegekben szereplő szóalakok igen jelentős részét nem tudja a rendelkezésre álló morfológiai elemzőprogram elemezni, mert hiányzik az adatbázisából a szó töve. Ugyanakkor ezeknek az elemezhetetlen szóalakoknak a nagy része tartalmaz toldalékokat, ezért valamilyen módon ezeket is elemezni kellene. Az ismeretlen szavak elemzését általában valamilyen sztochasztikus tanuló eljárásból származó modellel próbálják megoldani, amelyet tanító korpuszon fejlesztenek ki. Ez a modell aztán még kiegészíthető külső információval is, mint például a szókezdő nagybetű megléte [2]. Ezek az eljárások azonban, még akkor is, amikor igen nagy mennyiségű annotálatlan tanító anyagot képesek használni [3], nehézkesen alkalmazhatók magyar nyelvre, elsősorban a sokféle és hosszú toldalékszekvenciákból adódó „kevés adat" (*sparse data*) probléma miatt.

A dolgozat ezért egy már létező morfológiai elemzőn alapuló szimbolikus ismeretlenszó-elemző eljárást mutat be, amelyet nagy korpuszból nyert statisztikai információt használó modell egészít ki, melynek segítségével a szimbolikus ismeretlenszó-elemző által generált hipotézistér hatékonyan szűkíthető. A dolgozat a következőképpen épül fel. A 2. rész rövid leírást ad a morfológiai elemzőről. A 3. rész a szimbolikus ismeretlenszó-elemzőt tárgyalja, míg a 4. részben bemutatjuk a tesztelés illetve modellépítés során használt adatokat. Az 5. rész a különböző paraméterekkel futtatott teszteket írja le, és bemutatja, hogyan lehetséges a szimbolikus ismeretlenszó-elemző által generált elemzések számát nagy korpuszból nyert szóalak és szufixum statisztika segítségével csökkenteni. Rövid összefoglalás zárja a dolgozatot a 6. részben.

2. A morfológiai elemző

Bár a morfológiai elemzés kulcsfontosságú a természetes nyelvfeldolgozásban, különösen agglutinatív jellegű nyelvek esetében, a morfológiai elemzők mint különálló nyelvfeldolgozó eszközök kevés figyelmet kaptak az irodalomban, és legtöbbjük kereskedelmi termék[1]. Az általunk alkalmazott Humor („High speed Unification Morphology") morfológiai elemző szintén egy kereskedelmi cég terméke [6]. Az elemző klasszikus „egyed-és-elrendezés" stílusú elemzést [7] végez. A bemenő szóalakot morfémák sorozatára bontja, ahol mindegyikhez egy felszíni alakot, egy lexikai alakot, illetve egy kategóriacímkét rendel. Az elemző reguláris szónyelvtannal rendelkezik, így egyszerű morfémalistaként adja meg a lehetséges elemzéseket, vagyis nem rendel belső szerkezetet az elemzett szóalakhoz. Az elemző kimenetét az 1. ábra illusztrálja.

```
analyser>lehetőségekben
lehetőség[S_FN]+ek[I_PL]+ben[I_INE]
lehető[S_MN]+ség[D=FN_PROP]+ek[I_PL]+ben[I_INE]
lehet[S_IGE]+ő[D=MN_MIF]+ség[D=FN_PROP]+ek[I_PL]+ben[I_INE]
lesz[S_IGE]=le+hető[D=MN_HATO]+ség[D=FN_PROP]+ek[I_PL]+ben[I_INE]
```

FN = főnév MIF = melléknévi igenév S_ = tő
MN = melléknév PROP = MN→FN képző I_ = rag
INE= inesszívusz HATO = modális képző D_ = képző
PL = többes szám

1. ábra. A morfológiai elemző kimenete.

A morfémákat '+' jel választja el, reprezentációjuk a lexikai alakkal kezdődik, melyet a kategória követ. Ha a felszíni alak különbözik a lexikaitól, az előbbi '='

[1] Érdemes megjegyezni azonban, hogy az utóbbi időben a komplex morfológiájú nyelvekre fejlesztett annotált nyelvi erőforrások megjelenése miatt egyre több figyelem jut erre a területre is (vö. a Xerox eszközöket [4], illetve egy független implementációért lásd pl. [5]).

jel után szerepel, egyébként nincs megadva. A kategóriacímkét a morféma morfológiai kategóriáját meghatározó prefixum előzi meg. Képzők esetén a képzett szó szófaját is megadja az elemzés.

Nyelvfeldolgozó feladatokban az elemző által szolgáltatott nagyon részletes elemzésre általában nincs szükség. Ezért egy szótövesítő eljárást kell alkalmazni, amely az adott szóalak tövét és inflexiós toldalékait azonosítja, oly módon, hogy az összetett szavak tagjai és a derivációs toldalékok a tő részeként szerepelnek és nem jelennek meg független elemként az elemzésben. Az 1. ábrában található elemzéseket a lemmatizáló egy elemzéssé vonja össze, ahogy azt a 2. ábra mutatja.

```
lemmatiser>lehetõségekben
lehetõség[FN][PL][INE]
```

2. ábra. A lemmatizáló kimenete.

3. A szimbolikus ismeretlenszó-elemző

Semmilyen robusztus és széles lefedettséget biztosító nyelvfeldolgozó eszközlánc nem tud hatékonyan működni olyan eljárás használata nélkül, amely a rendszer tudásbázisa által nem ismert nyelvi jeleket képes kezelni. Tipikus sztochasztikus szóalakszintű annotáló eszközök, pl. egyértelműsítők, jellemzően valamilyen beépített toldalékelemző statisztikai modellt alkalmaznak. Magyar nyelvre azonban ezek a modellek a magas toldalékvariancia miatt nem adnak jó eredményt [8]. Ezért a jelen ismeretlenszó-elemző rendszer egy parciális szimbolikus elemzőn alapul, amely a lehetséges lemma plusz toldalék szekvenciákról a statisztikai modelleknél intelligensebb hipotéziseket képes generálni.

Más megközelítésektől eltérően a szimbolikus ismeretlenszó-elemző által felhasznált adat nem nagyszámú szóalak elemzése feletti általánosítás eredménye [9]. Magyar nyelvben ugyanis a nyílt szóosztályok tagjainak lehetséges toldalékolt alakjai túlságosan nagyszámúak ahhoz, hogy kezelhetőek legyenek ilyen általánosítás megtételéhez. E helyett az ismeretlenszó-elemző adatbázisa a normál morfológiai elemző építésénél használt nyelvtani leírásnak a nyílt szóosztályok minden lehetséges tővégződésére való alkalmazásával készült.

Az ismeretlenszó-elemző a nyílt szóosztályok (főnév, ige, melléknév) minden inflexiós toldaléksorozatát azonosítani tudja, és néhány nagyon produktív derivációs toldalék is elemezhető. Az ismeretlen szóalakok jelentős része idegen szó, melyek nem követik a magyar kiejtés szerinti helyesírást, ezért bizonyos, az eredeti morfológiai elemzőben meglévő megszorításokat az ismeretlenszó-elemzőben ki kellett iktatni, illetve gyengíteni kellett (pl. magánhangzó-harmónia). Azon fonológiai és ortográfiai megszorítások, melyeknek ezen rendhagyó helyesírású

alakok is engedelmeskednek, részei maradtak az ismeretlenszó-elemző adatbázisának, és elemzéskor ellenőrződnek is.

Az elemző által megengedett igei alakok formája erősen korlátozott. Mivel a magyar igetövek osztálya zárt, minden új tőnek egyértelműen azonosítható végződése van, amely valamilyen produktív igeképzőt tartalmaz. Az elemző csak abban az esetben javasol igei elemzést, ha ilyen végződés kapcsolódik a (hipotetikus) tőhöz. Ez a lépés jelentősen csökkenti a lehetséges elemzések számát, de egyben feltételezi, hogy a morfológiai elemző ismerni a zárt tőosztály összes elemét.

Minthogy alapvetően ugyanazok az inflexiós toldalékok követhetik a főnévi és melléknévi (valamint számnévi) töveket, ezek csupán morfofonológiai alapon történő megkülönböztetése gyakorlatilag lehetetlen. Ezért az ismeretlenszó-elemző adatbázisában nem tettünk különbséget főnévi és melléknévi tövek között. A számnevek zárt osztályt alkotnak, így a számnévi szufixumok sem kerültek be az adatbázisba. Azokban a (ritka) esetekben, ahol egyértelműen azonosítható a melléknévi toldalék, az elemző természetesen felismeri a helyes tőkategóriát, egyébként minden főnévi tövet ajánló elemzés egyben melléknévi tövet tartalmazó elemzésként is tekinthető. A főnévi kategória később felülírható, ha a szóalak melléknévnek bizonyul. Az ismeretlenszó-elemzőnek a lemmatizáló formátuma szerinti kimenetét a 3. ábra illusztrálja.

```
guesser>Torgyán
Torgyán[FN] [NOM]
Torgyá[FN] [SUP]
Torgya[FN] [SUP]
Torgy[FN] [PSe3] [SUP]
```

3. ábra. A szimbolikus ismeretlenszó-elemző kimenete.

4. Az adatok

Az ismeretlenszó-elemző eszközlánc átfogó teszteléséhez, illetve a tő- és szufixumeloszlások statisztikai modelljeinek felépítéséhez a Magyar Nemzeti Szövegtár [10] teljes 150 millió szavas anyaga szolgált nyelvi erőforrásul. A szöveg minimális előfeldolgozáson, tokenizáláson esett át, a speciális tokenosztályok külön kezelése nélkül. Ezt az „először nézzük, milyen morfológiai információt hordoz egy token" megközelítést a magyarban az indokolja, hogy a legkülönfélébb típusú tokenek, mint például rövidítések, tulajdonnevek, címek, tisztségek mind toldalékolhatók, ezért a speciálisan ezek kezelésére kifejlesztett nyelvfeldolgozó moduloknak is hozzá kell férniük a morfológiai információhoz.

A korpusz anyaga gyakorisági lista alakjában szolgált a morfológiai elemző (lemmatizáló) (ME) bemenetéül. Az 1. táblázat tartalmazza a morfológiai elemzés főbb adatait. Az ismeretlen alakokat ezután az ismeretlenszó-elemző dolgozta

1. táblázat. A morfológiai elemzés összefoglaló adatai.

Egységek	ME által elemzett	Ismeretlen	Összesen
Szóalak típus	2.222.280 (69.06%)	995.396 (30.94%)	3.217.676
Szóalak token	125.319.357 (95.50%)	5.907.372 (4.50%)	131.226.729

fel, amely minden egyes az ME által elemzetlenül hagyott alakhoz hozzárendelte a lehetséges elemzésük listáját. Az ismeretlenszó-elemző összesen 2.360.845 elemzést adott meg a 995.396 szóalakhoz, ami 2,37 elemzés/token átlagnak felel meg. Ez az érték jelentősen magasabb, mint az ME hasonló értéke (ahol 3.065.988 elemzés tartozott 2.222.280 szóalakhoz, 1,38 elemzés/token átlaggal). A különbségnek alapvetően két oka van: egyrészt néhány az ME-ben jelenlévő megszorítás az ismeretlenszó-elemzőből ki lett iktatva az idegen szavak elemzésének elősegítése miatt, másrészt a lemmatizáló gyakran összevon elemzéseket, melyeket az ismeretlenszó-elemző nem. Az utóbbi ugyanis megpróbál minél több derivációs toldalékot és ezen keresztül minél több tövet azonosítani, hogy az elemzések rangsorolását és értékelését végző statisztikai módszerekhez kimerítő alapadatokat szolgáltathasson. Érdemes megjegyezni, hogy amennyiben a lehetséges igei elemzések nem lennének ilyen mértékben korlátozva, illetve a melléknévi elemzés is alapesetben bekerülhetne a lehetséges elemzések közé, a fenti 2,37-es átlag megközelítené az 5-öt.

5. Az elemző tesztelése és kiértékelése

A szimbolikus ismeretlenszó-elemző által generált hipotézisteret természetesen érdemes szűkíteni a valószínűtlen elemzések kizárásával illetve alacsonyra rangsorolásával. Ezzel kapcsolatban releváns információ nyerhető például a korpuszban található toldalékszekvencák eloszlásából, melynek alapján többféle tesztmodellt is lehet vizsgálni.

5.1. Tesztmodellek

Az *1a.*-val jelölt modellben a preferált elemzés kiválasztása az ismeretlenszó-elemző által javasolt tőnek a korpuszban mért előfordulási gyakorisága alapján történt. Tehát az az elemzés számított a helyesnek, ahol az elemzéshez rendelt szótő a legtöbbször fordult elő mint független szóalak a korpuszban. A gyakorisági adatok a szóalakok kisbetűsített formája alapján lettek kiszámítva. A 4. ábrában látható az ismeretlenszó-elemző kimenete, ahol az elemzések a tő gyakorisága szerint vannak súlyozva.

Az *1b.* modell az előző kissé módosított változata, amennyiben egy szűrő ebben a modellben kizárt bizonyos elemzéseket, mielőtt azok az *1a.*-ban használt mérték (egyszerű tőgyakoriság) szerint rendezve lennének. A szűrő az alábbi

```
19957   Torgyán     Torgyán[FN][NOM] (19957)
                    Torgy[FN][PSe3][SUP] (0)
                    Torgyá[FN][SUP] (0)
                    Torgya[FN][SUP] (0)
1635    mindenképp  minden[FN][_KEPP] (175547)
                    mindenképp[FN][NOM] (1635)
598     Monde       Mond[FN][PSe3][NOM] (6792)
                    Monde[FN][NOM] (598)
```

4. ábra. Az ismeretlenszó-elemző kimenete az 1a. modellben.

módon működik: amennyiben az ME az ajánlott elemzéshez tartozó tövet egyébként tudta elemezni, de ezen elemzések között nincs olyan kategóriájú, amit az ismeretlenszó-elemző tulajdonított a javasolt tőnek (pl. a tőnek az ME szerint ige a kategóriája, viszont a javasolt elemzés főnévi kategóriát adna), akkor a kérdéses elemzést a szűrő kizárja. Az 5. ábra mutatja az ismeretlenszó-elemző szűrt és rangsorolt kimenetét. A *mond* alak igei tő az ME szerint, ezért a főnévi javasolt elemzést a szűrő kizárta.

```
19957   Torgyán     Torgyán[FN][NOM] (19957)
                    Torgy[FN][PSe3][SUP] (0)
                    Torgyá[FN][SUP] (0)
                    Torgya[FN][SUP] (0)
1635    mindenképp  minden[FN][_KEPP] (175547)
                    mindenképp[FN][NOM] (1635)
598     Monde       Monde[FN][NOM] (598)
```

5. ábra. Az ismeretlenszó-elemző kimenete az 1b. modellben.

A *2.* modell szintén figyelembe veszi az ME által szolgáltatott elemzéseket, de az *1b.*-ben alkalmazott szűrőn túl a kompatibilis elemzések tőkategóriája az ME által javasoltra íródott felül. A rangsorolás alapjául ebben a modellben nem az egyszerű tőalak gyakorisága szolgált, hanem a javasolt tő gyakorisága az ME általi elemzésekben. Azoknál a töveknél, amelyeket az ME nem elemzett, az előző modellekhez hasonlóan a szóalakgyakoriság maradt a mutató. A feltételezés a *2.* modell mögött az, hogy az ME számára ismeretlen szóalakok sokszor nem azért maradnak elemzetlenül, mert tövük hiányzik az ME adatbázisából, hanem vagy paradigmahiba van az ME-ben, vagy pedig az adott alak ortográfiája a tő–szufixum határon nem követi a szokásos eljárást (pl. kötőjel szerepel olyan helyen, ahol egyébként nem szokás, vagy fordítva.). A 6. ábra illusztrálja a *2.* modell kimenetét. A *mond* szótő ki van szűrve, és a *minden* tő gyakorisága megváltozott az előző modellekhez képest.

```
19957   Torgyán       Torgyán[FN][NOM] (19957)
                      Torgy[FN][PSe3][SUP] (0)
                      Torgyá[FN][SUP] (0)
                      Torgya[FN][SUP] (0)
1635    mindenképp    minden[FN|NM][_KEPP] (216310)
                      mindenképp[FN][NOM] (1635)
598     Monde         Monde[FN][NOM] (598)
```

6. ábra. Az ismeretlenszó-elemző kimenete a 2. modellben.

A *3.* modell olyan hasonlósági mértéket használ fel, amely az ismeretlenszó-elemző által javasolt tövek hasonlóságát próbálja megragadni az adott kategória jellemző töveihez (vagyis a *főnéviség, igeiség stb.* mértékét). Ennek érdekében kiszámoltuk az ME által elemzett összes szóalak tövéhez kapcsolódó toldalékok eloszlását, és ezen eloszlásokat tőkategóriánként tároltuk. Megszámoltuk azon elemzéseket, melyek egy adott tőkategóriával kezdődtek, és ezeket az értékeket elosztottuk az adott kategória összes előfordulásával. Így minden kategóriára (C') kaptunk egy normalizált eloszlást ($H(C')$). Ugyanezt az eljárást ismételtük meg az ismeretlenszó-elemző elemzéseire is, majd a javasolt tövek kategóriájának eloszlását ($H(S_C)$) összehasonlítottuk a kompatibilis tövek teljes eloszlásával ($H(C')$), és kiszámoltuk a két eloszlás abszolút különbségét ($AD(C', S_C)$). Ez a különbség egy 0 és 2 közötti szám ($AD(C', S_C) \in [0, 2]$); 0, ha a két eloszlás azonos és 2, ha egyáltalán nincs közös toldaléksorozat. A hipotetikus tő+kategória elemhez rendelt, a C' eredeti tövekhez való hasonlóságot kifejező mérték pedig a következő[2]: $SM(C', S_C) = \frac{2-AD(C', S_C)}{2}$. Ezután egy C' kategóriájú tövet tartalmazó elemzéshez rendelt mérőszám (OM) a C kategóriájú tő gyakorisági értékének ($F(S_C)$) és a hasonlósági mértéknek a szorzata: $OM(C', S_C) = SM(C', S_C)F(S_C)$.

Mint az 5.2. részben látható, ez a mérőszám nem bizonyult különösebben hatékonynak. Ez egyrészt a lemmatizáló és az ismeretlenszó-elemző működése közötti különbségből adódhat, ugyanis pl. a lemmatizáló általában nem ad vissza nominatívuszi főnév elemzést, ha a kérdéses szóalak derivációs toldalékra végződik, míg az ismeretlenszó-elemző igen, ezért pl. a nominatívuszi főnevek eloszlása jelentékenyen különbözik, ez pedig a melléknévi tőválasztást preferálja, ami sok hibához vezet. Másrészt azonban további vizsgálat szükséges annak érdekében, hogy milyen egyéb okok játszhatnak szerepet, illetőleg milyen más hasonlósági értékkel lenne érdemes számolni. A *3.* modell kimenetét a 7. ábra mutatja.

5.2. Kiértékelés

Miután legjobb szándékunk ellenére sem találtunk általánosan elfogadott eljárást ismeretlenszó-elemzők teljesítményének kiértékelésére, az alábbi forgatókönyvet választottuk. Az ismeretlen szóalakok gyakorisági listájában megállapítottunk

[2] Lényegében ez az eljárás a két eloszlás különbségét az ún. L_1 normával méri.

```
19957  Torgyán      Torgyán[FN][NOM] (10100)
                    Torgyá[FN][SUP] (462)
                    Torgya[FN][SUP] (462)
                    Torgy[FN][PSe3][SUP] (218)
1635   mindenképp   mindenképp[FN][NOM] (679)
                    minden[FN][_KEPP] (6) 598
       Monde        Monde[FN][NOM] (338)
                    Mond[FN][PSe3][NOM] (58)
```

7. ábra. Az ismeretlenszó-elemző kimenete a 3. modellben.

egy (önkényes) küszöbértéket (10), amely előfordulás alatt nem vettük figyelembe az adott tokent, kizárandó a nagy számú olyan „hulladék" alakot, amihez legfeljebb az *egyéb* elemzés lenne rendelhető — ezek igen nagy méretű korpuszokban elkerülhetetlenek. A maradék listát felosztottuk 10 egyenlő gyakorisági tartományra, és mindegyikből véletlenszerűen választottunk 100 alakot. Az eredményként kapott 1000 szavas listán értékeltük a modelleket pontosság szempontjából.[3]

Annak mérésére, hogy a korpuszból nyert statisztikai adatok mennyiben javítják a szimbolikus ismeretlenszó-elemző teljesítményét, két viszonyító alapmodellt is kiértékeltünk. A *0a.* modell az egyenlő valószínűségűnek tekintett javasolt elemzések közül véletlenszerűen választott, míg a *0b.* modell mindig a nominatívuszi főnév elemzést adta. A teljesítményre vonatkozó értékek a 2. táblázatban találhatók.

2. táblázat. Az elemző teljesítménye különböző statisztika modellekben.

	Modell		Teljesítmény	
			Típus	Token
korpuszadat nélkül	0a	véletlen választás	69.76%	53.39%
	0b	FN alanyeset	78.09%	88.72%
korpuszstatisztikával	1a	tőfrekvencia	84.18%	91.89%
	1b	szűrt tőfrekvencia	84.61%	92.73%
	2	hibrid frekvencia	84.61%	92.69%
	3	eloszlás összehasonlítás	84.29%	91.85%

[3] Ez egy egyszerűsített értékelés, amelyben a modellek minden alakhoz egy elemzést választanak, így külön *fedés* és *pontosság* értékek itt nem számolhatók. Ha a leírt eljárást szófaji egyértelműsítés kontextusában lexikális valószínűség értékek indukciójához használjuk — ez a jelen dolgozat témájának egyik lehetséges továbbfejlesztése —, akkor a különböző értékek már számolhatók.

Minthogy az alanyesetű főnév nagyon gyakori az ismeretlen szavak között, már a *0b.* alapmodell is meglehetősen jól teljesített. Ugyanez a tendencia kiegészítve azzal, hogy az igék leggyakrabban jelen idő, egyes szám 3. személy kijelentő módban szerepelnek, eredményezi a minimális statisztikával támogatott *1a.* modell jó eredményét. A tövekre vonatkozó szűrés tovább csökkenti a hibák számát az *1b.* modellben. A 3. modell viszonylag gyenge teljesítményét az előző részben már említettük.

Az eredményeket sztenderd metodológia hiányában kissé nehézkes más hasonló próbálkozások eredményével összevetni. Alegria et al. [11] egy szófaji egyértelműsítő rendszer általános teljesítményét adja meg, amely ismeretlenszó-elemzést is használ (93%), míg Chanod és Tapanainen [12] az ittenihez hasonló kiértékelés szerint 85 %-os pontosságot ér el, bár meglehetősen szűk elemzési kódkészlettel (az általunk használt készlet több ezer lehetséges kódot tartalmaz).

6. Összefoglalás

Egy olyan ismeretlenszó-elemző rendszer kifejlesztését mutattuk be, amely szimbolikus megszorításokon alapuló részleges elemzőt egészít ki nagy korpuszból nyert olyan statisztikai információval, melynek segítségével az első lépésben előállított hipotézistér a kívánt mértékűre szűkíthető. A szimbolikus elemző és a statisztikai szűrő együttesét alapvetően két feladat ellátására látjuk alkalmasnak. Az egyik feladat a folyó szövegben előforduló ismeretlen szóalakok on-line elemzése és egyértelműsítése, a másik a morfológiai elemző adatbázisának bővítése, illetve javítása (off-line adatgyűjtés).

Az első feladat esetében a konkrét szóalakhoz egyetlen olyan elemzést kell kiválasztani, amely a szó tövét és morfoszintaktikai jegyeit (a tő és az inflexiós toldalékok kategóriáját) leírja. A másik feladat megoldásához olyan töveket kell a korpuszból kiválasztani, és a kategóriájukat megfelelően azonosítani, illetve esetleges egyéb megjósolhatatlan morfológiai tulajdonságaikat a korpuszban szereplő toldalékolt alakjaik segítségével megállapítani, amelyeket érdemes lenne a morfológiai elemző adatbázisába felvenni. A rendszer ezen két irányban történő alkalmazása jelenleg folyó kutatás tárgyát képezi.

Hivatkozások

1. Hajič, J.: Morphological tagging: Data vs. Dictionaries. In: Proceedings of ANLP-NAACL Conference, Seattle, Washington, USA (2000) 94–101
2. Weischedel, R., Meteer, M., Schwartz, R., Ramshaw, L., , Palmucci, J.: Coping with ambiguity and unknown words through probabilistic models. Computational Linguistics 19 (1993) 359–382
3. Cucerzan, S., Yarowsky, D.: Language independent minimally supervised induction of lexical probabilities. In: Proceedings of the 38th Annual Meeting of the Associations for Computational Linguistics, Hong Kong (2000) 270–277
4. Karttunen, L.: Applications of finite-state transducers in natural language processing. In: Proceedings of CIAA-2000. Lecture Notes in Computer Science, Springer Verlag (2000)

5. Alegria, I., Aranzabe, M., Ezeiza, A., Ezeiza, N., Urizar, R.: Using finite state technology in natural language processing of Basque. In: Proceedings of the Conference on Implementations and Applications of Automata, Pretoria (2001) 2–12
6. Prószéky, G., Kis, B.: Morpho-syntactic parsing of agglutinative and other (highly) inflectional languages. In: Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics, College Park, Maryland, USA (1999) 261–268
7. Hockett, C.F.: Two models of grammatical description. Word 10 (1954) 210–234
8. Oravecz, Cs., Dienes, P.: Efficient stochastic part of speech tagging for Hungarian. In: Proceedings of the Second International Conference on Language Resources and Evaluation, Las Palmas (2002) 710–717
9. Daciuk, J.: Finite state tools for natural language processing. In: Proceedings of the COLING 2000 workshop Using Toolsets and Architectures to Build NLP Systems, Luxembourg, Luxembourg (2000) 34–37
10. Váradi, T.: The Hungarian National Corpus. In: Proceedings of the Second International Conference on Language Resources and Evaluation, Las Palmas (2002) 385–389
11. Alegria, I., Aranzabe, M., Ezeiza, A., Ezeiza, N., Urizar, R.: Robustness and customisation in an analyser/lemmatiser for Basque. In: Proceedings of the LREC-2002 Workshop on Customizing Knowledge in NLP Applications, Las Palmas (2002)
12. Chanod, J.P., Tapanainen, P.: Creating a tagset, lexicon and guesser for a French tagger. In Tzoukermann, E., Armstrong, S., szerk.: From Texts to Tags: Issues in Multilingual Language Analysis: Proceedings of the ACL SIGDAT Workshop, Geneva (1995) 58–64

Development of a morphological guesser for Hungarian

Attila Novák[12], Viktor Nagy[2], and Csaba Oravecz[2]

[1] MorphoLogic Ltd., Budapest
[2] Research Institute for Linguistics, Budapest
{novak,nagyv,oravecz}@nytud.hu

Keywords: guesser, morphological analysis, similarity of distributions, disambiguation

Highly inflectional/agglutinative languages like Hungarian typically feature possible word forms in such a magnitude that machine learning methods which are most often used in NLP and rely on training data almost always face the problem of data sparseness. This problem cannot simply be tackled by independently preparing huge morphological dictionaries; these will grow to sizes unmanageable to any efficient application. A hand-on solution to this problem is to apply a comprehensive morphological analyser, which works in tandem with a base form lexicon and has the capability of analysing all inflectional, productive derivational and compounding phenomena and is also capable of doing base form reduction. Although essentially being a symbolic tool, such an analyser can be efficiently utilized even in a stochastic NLP environment.

Independently of the type of the source that provides the lexical information, morphological processing of huge corpora inevitably faces the problem of a large number of unknown word forms. For the symbolic analyser, this means that the particular base form is not listed in the analyser's lexicon so its derivatives cannot be analysed. (Many of these are of foreign origin with irregular orthography, which poses a special problem in Hungarian where suffixation is primarily determined by the phonological shape of the stem which is not reflected by the orthographic form of these words in any consistent way.) In order to cope with the problem of unknown words in unconstrained corpora, generally some stochastic method is used based on suffix models built from training corpora and aided by some external information like capitalization. However, the direct application of these models, even when supported by information from very large corpora, is debatable in the case of languages like Hungarian.

As far as the external information is concerned, its use does not contribute significantly to the knowledge of the guesser model. In Hungarian, almost any type of special tokens can get suffixation very productively, so e.g. the information that some form is capitalized so it is probably a proper noun is uninformative for the system: it does not deliver the essential information on the base form and the details of suffixation attached to it.

As for the stochastic suffix models, it is worth noting, that with the application of a morphological analyser there is an important difference in the nature of the unknown word problem: we have to handle word forms unknown to the morphological analyser and not word forms not found in training corpora. Fixed and variable length suffix models are based on annotated training corpora and in Hungarian face the same data sparseness problem as any other pure stochastic NLP method. Models built upon unannotated corpora of potentially unlimited in size introduce a huge search space in our case which is difficult to manage computationally. In addition, when using these models in a practical application, a fairly strict limit must be set on the maximal length of the suffixes to be considered. But in Hungarian, due to the agglutinating nature of the language, very long inflectional suffix sequences do occur, which poses an inherent problem for all purely stochastic suffix models.

The paper present a combined method for unknown word guessing featuring symbolic constraints and statistical information. The former is embodied in a partial word form analyser (guesser) which generates hypotheses on possible lemma-plus-suffix sequences along with properties which can be inferred for the lemma from the suffix sequence. This hypothesis space is then pruned using statistical information concerning word form and suffix sequence distribution gathered from a 150 million word corpus analysed by the morphological analyser. The morphological knowledge built into the symbolic guesser is directly derived from the linguistic description used for the creation of the morphological analyser.

Since unknown words in general tend to belong to productive inflectional and derivational paradigms the hypothesis space can effectively be reduced in the first place by considering only these paradigms in the partial analysis. To associate weights to the outputs of the partial analyser and to exclude improbable analyses several models are developed based on the statistical information from the corpus. Measures evaluated range from simple relative stem frequency to similarity measures like L1 norm between stem/suffix distribution proposed for the unknown forms and stem/suffix distribution of the known word forms. Evaluation is carried out from two perspectives: with respect to the extension of the lexicon of the morphological analyser with base forms gained from the unknown word analysis, and with respect to the induction of lexical probabilities for the unknown forms; these probabilities are used by a part-of-speech tagging system especially well suited and robust for morphological processing of unconstrained Hungarian language data.

Új szakkifejezések keletkezésének vizsgálata számítógépes szakirodalmi adatbázisok segítségével a molekuláris genetika szaknyelvében

Solymosi Mária

SZTE Idegennyelvi Központ

msolymosi@freemail.hu

Kulcsszavak: új szakkifejezések keletkezése; molekuláris genetikai korpusz; szógyakoriság; idézet elemzés

Absztrakt.
Az informatikához hasonlítható robbanásszerű változást eredményez a molekuláris szemléletmód térnyerése az élettudományokban. A tudományágban állandóan megjelenő új jelenségek váltják ki a molekuláris genetika terminológiájának változását és növekedését.
A jelen dolgozat az új szakkifejezések kialakulását kívánja vizsgálni, azt a folyamatot, ami lehetővé teszi, hogy egy-egy fogalom a szakmai közvéleményben széles körben ismertté váljon.
Az információs technológia adott fejlettségi szintje már lehetővé teszi, hogy a témát átfogó vizsgálatnak vessük alá. A jelen tanulmány során internetes adatbázist felhasználva kerül sor a genetikai korpusz megjelenítésére és a szakma által újnak ítélt szakkifejezések kiválasztására. Egy új szakkifejezés vizsgálatakor az idézetelemzés segítségével kíséreljük meg azonosítani azt a közleményt, amiben az új tudományos fogalmat megnevező szó már megtalálható és a rá vonatkozó hivatkozások száma ugrásszerűen megnő, így a fogalom közismertté válik, ami a gyakorisági vizsgálat módszerével nyomon követhető.
A jelen dolgozatban egy új kutatási módszert kívánunk bemutatni, amely a változó molekuláris genetikai szaknyelv fejlődésének tendenciáin keresztül más szaknyelvi terminológiák szabályszerűségeire is enged következtetni.

1. Bevezetés

1.1. A terminológia-kutatás időszerűsége

A szaknyelvi terminológiai kutatások iránti megnövekedett érdeklődés a nagy sebességgel változó, állandó mozgásban lévő információs társadalom kihívásaival indokolható. A WEB-et böngészve, számtalan hazai és nemzetközi kutatóközpont létrejöttéről és működési elveiről szerezhetünk tudomást (1). Ezen intézmények, alapítványok közös vonása a nyitottság, gyors felhasználhatóságra törekvés, rugalmas adaptáció a világ más részein már működő rendszerekhez, trendekhez. Világosan

fogalmazzák meg azokat a célokat, amelyeket a magyar felhasználók érdekében mihamarabb meg kell valósítani. A nyelv a maga komplexitásával az interdiszciplináris kutatások egyik kulcsfontosságú területévé vált, a tudásalapú társadalom nélkülözhetetlen eszköze lett. A nyelv felől közelítve az információ gyors áramlása az ekvivalenciákra épülő terminológiai adatbázisok használatával nagymértékben elősegíthető. Az alkalmazott nyelvészet terminológiai kutatásokkal foglalkozó ága hozzájárul az információs társadalom stratégiai feladatainak megoldásához. Az egymást értő, sikeres kommunikáció megvalósulásának elengedhetetlen eszközei az elektronikus szótárak, tezauruszok, terminológiai adatbázisok, másképpen fogalomtárak. Ezen belül kiemelt helyet foglal el a szakma-specifikus terminológiai adatbázisok megalkotása. Nyomtatott és elektronikus formában egyaránt elérhető szótárak és fogalomtárak teszik lehetővé a gyors, célirányos szakmai kommunikációt.

1.2. A témaválasztás célja

A jelen dolgozat a molekuláris genetika terminológiájának sajátos tendenciáit kívánja vizsgálni, a szakkifejezések megjelenésének körülményeit és azt a folyamatot, ami lehetővé teszi, hogy egy-egy fogalom a szakmai közvéleményben széles körben ismertté váljon. Az információs technológia adott fejlettségi szintje, az adatbázisok gyors elérhetősége már lehetővé teszi, hogy a témát átfogó vizsgálatnak vessük alá, míg hagyományos módszerrel, a folyóiratokban megjelenő szakkifejezések történeti tanulmányozása igen hosszú időt venne igénybe. A jelen interdiszciplináris kutatás több tudományterület lehetőségét kívánja felhasználni. A munkát végző kutató alkalmazott nyelvész, a vizsgálat tárgya a molekuláris genetika, az eszköze pedig az informatika eszköztára. Ez a kutatási helyzet, az állandó szakmai kontroll igénye számos nehézséget rejt magában. A jelen alapkutatás a felsőoktatásban folyó szaknyelvoktatás egyik célkitűzésének, a szakterminológia megismertetésének gyakorlati célját kívánja támogatni.

2. A kutatás módszere

A kutatás alapötletét a molekuláris genetikai terminológia használata iránti szükséglet és az új terminusok sajátossága váltotta ki. Módszertanilag olyan megközelítést, eljárást dolgoztam ki, ami a kutatási feladat megoldására alkalmasnak ígérkezik. A jelen dolgozatban ezt kívánom bemutatni. Egy tervezett nagyobb kutatási feladat keretein belül nagyobb számú példaelemzés elvégzésével a módszer új összefüggések bemutatására válhat alkalmassá.

2.1. Az vizsgált genetikai terminus kiválasztása

A vizsgálat tárgya az interneten elérhető számos genetikai korpusz közül véletlenszerűen kiválasztott 656 szócikkből álló glosszárium (2).
Az első lépés a vizsgálandó kifejezések kijelölése volt. Genetikus szakemberek jelölték meg az általuk újnak tartott kifejezéseket a kiválasztott glosszáriumban.

Definiálni kellett azonban az „új" fogalmát. Az első megközelítésben 1953-at, a DNS felfedezésének dátumát, a molekuláris genetika születésének időpontját jelöltük ki. Az előkísérletek azt mutatták, hogy némely kifejezés közben már jelentésmódosuláson ment át, így nem tesz eleget az „új" kritériumának (például a „conjugation" terminus, 1907-től a kémiában és biokémiában használatos, majd az 1960-as évek közepétől más jelentéssel átveszi a genetika). Így további szűkítésre volt szükség. Az időhatárt az1970-80-as évekre csökkentettük. Ez már szerencsés választásnak bizonyult, a terminus a keletkezésekor szerzett alapjelentését a mai napig megőrizte.
A dolgozatban vizsgált terminus a reverse transcriptase. A fent említett glosszárium szócikkei aktívak (3), így az adott címszó alatt a következő definíciót és a vele összefüggésbe hozható szakkifejezéseket találhatjuk (1 ábra).

Reverse transcriptase	An enzyme that catalyzes the synthesis of DNA from an RNA template.

Related Terms:

Enzyme	A protein that acts as a catalyst, speeding the rate at which a biochemical reaction proceeds but not altering the direction or nature of the reaction.
Deoxyribonucleic acid (DNA)	The molecule that encodes genetic information. DNA is a double-stranded molecule held together by weak bonds between base pairs of nucleotides. The four nucleotides in DNA contain the bases: adenine (A), guanine (G), cytosine (C), and thymine (T). In nature, base pairs form only between A and T and between G and C; thus the base sequence of each single strand can be deduced from that of its partner.
Ribonucleic acid (RNA)	A chemical found in the nucleus and cytoplasm of cells; it plays an important role in protein synthesis and other chemical activities of the cell. The structure of RNA is similar to that of DNA. There are several classes of RNA molecules, including messenger RNA, transfer RNA, ribosomal RNA, and other small RNAs, each serving a different purpose.

1.ábra

A definíció: a reverse transcriptase egy enzim, DNS polimeráz, ami mintaként RNS-t használ, RNS függő DNS polimeráz. Jelentése: míg az élővilágra általánosan jellemző információ áramlás iránya: DNS→ RNS→ fehérje, (ez a stratégia „centrális dogmaként" vonult be a tudományba (4), addig az RNS vírusokra jellemző stratégia: RNS→ DNS→ RNS→ fehérje. Ez egy szabályt erősítő kivétel.

2.2. A terminus gyakorisági vizsgálata

A kiválasztott reverse transcriptase terminus gyakoriságát a HighWire helyről a Medline adatbázisban (5) vizsgáltam. A Medline több mint 4500 szakmai folyóiratra, e sorok írásakor 12,797,216 cikkre épülő bázis, ezt a HighWire még 618,551 teljes terjedelmű cikkel egészíti ki. A kulcsszó és évszám megjelölésével megkaptam az adatbázisban az adott évekre vonatkozó gyakorisági mutatókat. 2003-tól időben visszafelé haladva végeztem az adatgyűjtést, amíg el nem jutottam az első megjelenés dátumáig. Az adatokat egy saját szerkesztésű táblázatba rendezve nyomon követhetővé vált a reverse transcriptase terminus gyors fejlődése (1. táblázat). Ugyanakkor feltételezhető, hogy a fogalom már valószínűleg korábban is ismert volt és más néven a szakma használta, de az új terminus hivatalos elfogadása csak

nemzetközi jelentőségű tudományos közlemény hatására következett be. Az RNA dependent DNA poymerase terminus 1959-ben 2 közleményben jelent meg, a mai napig ismert a szakmában, de a frappánsabb, egyszerűbb kifejezés az adatokat alapul véve 1993-ban átvette a vezető szerepet (1.táblázat).

évszám	*Reverse transcriptase* találat	*RNA depDNApolymerase* találat
2003*	10396*	7853*
2002	12413	8189
2001	12104	8654
2000	10858	8061
1999	9751	7162
1998	6710	5867
1997	4384	3861
1996	3176	2617
1995	2462	2110
1994	1914	1692
1993	1328	1353
1992	1040	1147
1991	787	811
1990	693	584
1989	489	532
1988	411	436
1987	332	312
1986	217	246
1985	188	232
1984	152	201
1983	140	185
1982	116	192
1981	121	157
1980	125	203
1979	143	206
1978	122	192
1977	115	206
1976	106	208
1975	105	269
1974	64	161
1973	41	111
1972	29	122
1971	10	86
1970		49
1969		46
1968		38
1967		28
1966		29
1965		19
1964		19
1963		11
1962		9
1961		2
1960		1
1959		2

* még nem lezárt év

1. táblázat

A kutatás további részében a Medline adatbázisban megjelenítettem a reverse transcriptase 1971-re vonatkozó lapját, ahol a terminust először használó 10

közlemény bibliográfiai adatai fellelhetők (6). A közlemények a szakma legnevesebb folyóirataiban jelentek meg (Nature, Nature New Biology, Lancet, Developmental Biology, Proc. Nat. Acad. Sci. USA, Science), megerősítve azt a tényt, hogy egy szakkifejezés akkor válik ismertté, amikor valamelyik nemzetközileg elismert szakmai lap közli. A 10 közlemény 1971 május és december között jelent meg szorosan egymás után. A szakma felfokozott érdeklődése a téma iránt jelzi a reverse transcriptase jelentőségét.

2.3 Idézetelemzés

Ezután a 10 közlemény behatóbb vizsgálata következett. A közlemények szakkönyvtárakban elérhető nyomtatott példányai egy későbbi nyelvi szempontú elemzés alapjául szolgálnak. A jelen dolgozatban az interneten elérhető adatbázisok terminológiai kutatás céljára történő felhasználhatóságát kívánom a vizsgálat középpontjába állítani, a reverse transcriptase terminus keletkezésének körülményeit nyomon követni. Az ISI Web of Science adatbázisban (7) a hivatkozási index (cited reference search) vizsgálata a következő eredményre vezetett. A szerző és év keresésekor azonosíthatóvá vált a közlemény és a keresés időpontjáig rá történt hivatkozások száma. A jelen esetben fontos időrendiség miatt a táblázatban a számozás alulról indul. (2. táblázat)

1971		**1970**	
Név	Idézettségi index	Név	Idézettség
10 Mollig	140		
9 Gallo *Review*	-	**Temin** **Baltimore** **Gallo**	**1412** **1390** **261**
8 Schlom	-		
7 Kotler	-		
6 Goodman	191		
5 Scolnic	-		
4 Gallo	28		
3 Sirtori *Letters to the Editor*	-		
2 *Editorial:* Happy birthday, ...	-		
1 *News and Views:* Reverse transcriptase	-		

2. táblázat

Az 1971-ben megjelent 10 közlemény közül, amelyik már a címében is tartalmazza a reverse transcriptase-t, 4 összefoglaló méltatás (1.2.3.9.). Ezek áttekintik a témát, így további referenciával szolgálnak. Az ott közzétett hivatkozások nyomon követésével értékes információhoz jutunk a terminus keletkezésével kapcsolatban. A jelen példában az összesítés alapján arra következtethetünk, hogy a reverse transcriptase terminus 1971-ben született. Ennek valószínűleg nyelvi bizonyítékai is vannak a közleményben. Ugyanakkor a vele azonos jelentésű RNA-dependent DNA Poymerase terminust találjuk Temin és Baltimore 1970-ben, a Nature-nek ugyanabban a számában megjelent cikkeiknek címében és Gallo ugyancsak 1970-es Nature cikkében (8). Ezek a közlemények olyan magas idézettséget értek el, ami alapján feltételezhetjük, hogy az ott közölt eredmények a mai napig kivívják a szakma elismerését.

Összegzés

Az új szakkifejezések keletkezésének vizsgálata során eddig elvégzett anyaggyűjtés és példaelemzés arra enged következtetni, hogy a téma számos, izgalmas kérdést tartogat a kutató számára. A váratlanul felmerülő problémák sok esetben teljesen új kutatási helyzet elé állítják az elemzőt. A téma vizsgálata rugalmas és széles merítésű háttér bázist és eszköztárat igényel, amit az interneten elérhető adatbázisok reményeink szerint biztosítani tudnak.

Hivatkozások

http://eisz.om.hu (Oktatási Minisztérium, Elektronikus információszolgáltatás, Web of Science)
http://www.ittk.hu (Információs Társadalom-és Trendkutató Központ)
http://www.scriptum.hu (Scriptum Informatika Rt.: lexikográfiai alkalmazások, kutatásfejlesztés)
http://www.linux.infoterm.org (Infoterm, International Information Centre for Terminology; Termnet, International Network forTerminology)
http://www.oszk.hu (Országos Széchényi Könyvtár)
http://hal.weihenstephan.de/genglos/asp/genreq.asp?list=1
Birgid Schlindwein's Hypermedia Glossary Of Genetic Terms Alphabetical list of all 656 items of the glossary
http://hal.weihenstephan.de/genglos/asp/genreq.asp?list=1 Birgid Schlindwein's Hypermedia Glossary Of Genetic Terms Alphabetical list of all 656 items of the glossary
News and Views: Central Dogma Reversed.
Nature 226, 1198 (1970)
http://highwire.stanford.edu
http://highwire.stanford.edu
Az adatbázisban reverse transcriptase keresőszóval az 1971-es évben az alábbi 10 közleményt találhatjuk.
K Molling, DP Bolognesi, H Bauer, W Busen, HW Plassmann, and P Hausen
Association of viral reverse transcriptase with an enzyme degrading the RNA moiety of RNA-DNA hybrids.
►Nat New Biol, Dec 1971; 234(51): 240-3.
RC Gallo
Reverse transcriptase, the DNA polymerase of oncogenic RNA viruses.
►Nature, Nov 1971; 234(5326): 194-8.
J Schlom and S Spiegelman
Simultaneous detection of reverse transcriptase and high molecular weight RNA unique to oncogenic RNA viruses.
►Science, Nov 1971; 174(11): 840-3.
M Kotler and Y Becker
Rifampicin and distamycin A as inhibitors of Rous sarcoma virus reverse transcriptase.
►Nat New Biol, Sep 1971; 234(50): 212-4.
NC Goodman and S Spiegelman
Distinguishing reverse transcriptase of an RNA tumor virus from other known DNA polymerases.
►PNAS, Sep 1971; 68(9): 2203-6.

7

EM Scolnick
"Reverse transcriptase" in higher cells.
►Dev Biol, Sep 1971; 26(1): 175-6
RC Gallo, PS Sarin, PT Allen, WA Newton, ES Priori, JM Bowen, and L Dmochowski
Reverse transcriptase in type C virus particles of human origin.
►Nat New Biol, Aug 1971; 232(31): 140-2
C Sirtori
Australia antigen, coronavirus, and reverse transcriptase in viral hepatitis.
►Lancet, Jul 1971; 2(7718): 261.
Happy birthday, reverse transcriptase?
►Nat New Biol, Jun 1971; 231(23): 161.
Reverse transcriptase in human milk virus.
►Nature, May 1971; 231(5298): 80.
http://eisz.om.hu (Oktatási Minisztérium, Elektronikus információszolgáltatás, Web of Science)
http://eisz.om.hu (Oktatási Minisztérium, Elektronikus információszolgáltatás, Web of Science)
E helyen a szerző neve és az év alapján kapjuk meg az idézetek számát.
Baltimore,D., Viral RNA-dependent DNA Poymerase.
Nature, 226, 1209 (1970)
Temin, H. M., and Mizutani, S., RNA-dependent DNA Poymerase in Virions of Rous Sarcoma Virus.
Nature, 226, 1211 (1970)
Gallo,R.C., RNA Dependent DNA Poymerase of Human Acute Leukaemic Cells.
Nature, 228, 927 (1970)

A study of emerging terminology in molecular genetics with the aid of internet databases

Solymosi Mária

SZTE Idegennyelvi Központ

msolymosi@freemail.hu

Keywords: emergence of new terminology; molecular genetic corpus; word frequency; citation index search

The introduction of a molecular approach in the life-sciences has resulted in a continuing revolutionary change in the terminology involved, which can only be compared to that in informatics. These developments in molecular genetic terminology have been triggered by the new phenomena that are constantly being discovered in this field. The present paper examines the occurrence of such new terminology and the process whereby certain expressions and concepts become widespread in the scientific community. The high level of development in informational technology allows a comprehensive study of this topic. Nevertheless, a search through the history of published terminology in scientific journals by traditional methods would be long-lasting and cumbersome. Accordingly, in this study an internet-available genetic glossary has been utilized to select the most up-to-date terms. These new terms were examined by using a database to detect their frequencies. This method helped reveal the first use and spreading of such genetic nomenclature. Further, a citation index search was carried out to identify those references which furnished significant information about the scientific importance of the terms in question. A retrospective citation search led to the first publication in which the term was applied. This paper aims to introduce a method based on informational technology in order to reveal tendencies in the rapidly developing terminology of molecular genetics, which might be relevant in other fields of sciences.

Főnévi csoport annotációja a CLaRK rendszerrel

Váradi Tamás

[3] MTA Nyelvtudományi Intézet
1068 Budapest, Benczúr u 33
varadi@nytud.hu

Kulcsszavak: felszíni szintaktikai elemzés, NP annotáció, lépcsős reguláris grammatika

Absztrakt. A magyar mondat szerkezetének leírásában kiemelt szerepet játszik a főnévi csoport. E dolgozat keretében beszámolunk arról a folyó munkáról, amely véges állapotú grammatika alkalmazásával megkísérli főnévi csoportok lehető legteljesebb felszíni leírását. Az ún. lépcsős reguláris grammatika (Abney 1996) kifejlesztése a CLaRK rendszerrel történt, melynek bemutatása szintén, melynek bemutatása szintén célja a jelen dolgozatnak.

1 Bevezetés

A dolgozat célja, hogy betekintést adjon a főnévi csoport automatikus felismerését célzó munkálatokba. A kutatások jelenleg is folynak, ezért az itt közzétett eredmények csak közbenső jelentésnek tekinthetők. A főnévi csoport annotációt szabályokra épülő rendszerben, a lépcsős reguláris grammatika módszerével (Abney 1996) végezzük. A fejlesztői keretrendszerül a CLaRK rendszert használjuk (Simov 2001), amely hatékonyan támogatja a kézi grammatikafejlesztést. A dolgozat felépítése a következő: az 2. részben ismertetjük a magyar főnévi csoport gépi feldolgozás szempontjából releváns sajátosságait, a 3. rész bemutatja a feldolgozott adatok szerkezetét és annotációjukat. Ezt követi a CLaRK rendszer rövid áttekintése a 4. részben, mely után ismertetjük a főnévi csoport felismerésére kifejlesztett szabályrendszer fő elveit. Az 6. rész tartalmazza magukat a szabályokat, melyek értékelését a 7. részben találjuk.

2 A kiinduló nyelvi tények rövid jellemzése

A magyar nyelvet közkeletű felfogás szerint szabad szórendű nyelvnek tekintik. Pontosabban fogalmazva, a magyarban a mondatszintű szintaktikai összetevők (szintagmák) viszonylag szabad sorrendben helyezkedhetnek el. Lényeges azonban látnunk, amint azt É. Kiss (1994) Brassai nyomán hangsúlyozza, hogy a mondatok szórendjét a topic-comment szerkezet határozza meg elsősorban, amelyet viszont a közlés

illetve a mondatokon átívelő szöveg kommunikációs sajátosságai szabnak meg. A szintagmákon belül az összetevők sorrendje kötött.

A viszonylag szabad szórendet a rendkívül gazdag alaktan teszi lehetővé, ugyanis a szintaktikai szerepeket a szintagmák főtagjának ragja jelzi. Ebből fakad az a sajátosság, hogy az egyszerű magyar mondatok döntő többségét egy ige és a körülötte található ragos főnévi csoportok alkotják. Esetragos főnévi csoportokkal fejezünk ki olyan viszonyokat, amelyeket más nyelvekben prepozíciós kifejezésekkel vagy határozószókkal fejtünk ki. Ez a tény ad kitüntetett jelentőséget a főnévi csoportok vizsgálatának.

A főnévi csoportok belső szerkezetének sajátosságaiból csak néhányat emelünk ki, amelyek megnehezíthetik az automatikus felismerést. Az első tény, amit megjegyezhetünk, hogy sajnos nincsenek olyan egyértelmű támpontok, amelyek minden helyzetben jelölnék a főnévi csoportok határait. A ragos főnevektől várhatnánk, hogy egyben a főnévi csoport jobb szélét is jelölnék de a birtokos és az igeneves szerkezetek miatt ez gyakran nincs így, másrészt a főnévi csoportból hiányozhat is maga a főnév, mely esetben a jelző veszi át a szerepét és egyben toldalékait. A főnévi csoport kezdetét egy determináns elem jelölheti ugyan, de ezek jelenléte még kevésbé feltételezhető, mint a főnévi fejé, másrészt a rekurzív beágyazódásból és az igenes szerkezetek bővítményeiből az is következik, hogy nem egyszerű feladat a determináns elem hovatartozását megállapítani.

Az igeneves szerkezetek elemzése különleges nehézséget jelent. A problémát az okozza, hogy a folyamatos vagy befejezett igenév (melynek szófaji besorolása szintén nem egyszerű feladat, hiszen az gyakran megkívánja a szintaktikai szerep elemzését is) olyan elem, amelyik gyakran hozza magával a bővítményei egész sorát mintegy beágyazott tagmondatot alkotva a főnévi csoporton belül. Egyéb nyelvekben a főnévi fejet követő prepozíciós szerkezettel fejezzük ki mindezt, itt tehát ugyanazzal a problémával találkozunk a magyar főnévi csoporton belül, amelyet a prepozíciós szerkezettel bíró nyelvekben a PP csatolás nehézségei címszó alatt tartanak számon.

3. Az adatok

A főnévi csoportok annotációját megelőzi a szöveg morfoszintaktikai elemzése. Ez arra a technológiára épül, amellyel a Magyar Nemzeti Szövegtár elemzett és egyértelműsített változata készült. A jelen kísérlethez az MNSZ morfoszintaktikai annotációjának némileg leegyszerűsített xml változatát használtuk. Az egyszerűsítés nem érintette a szavakhoz társított nyelvi elemzést. Minden szóalak (token) egy <w> elemen belül fordul elő és három attributum tartozik hozzá, melyek a lemmát, a morfoszintaktikai jellemzőt (msd) és a korpusz tag-et tartalmazzák.

A szintaktikai elemzés minőségét nagyban meghatározza a morfoszintaktikai annotáció és az egyértelműsítés pontossága. Az MNSZ annotációs rendszere alapvetően a HUMOR rendszer (Prószéky és Tihanyi 1996) jelkészletét használja, bár annak kimenetét további szűrésnek veti alá a párhuzamos elemzések kiszűrése és a lemma megállapítása céljából. Az egyértelműsítés pontossága eléri a 98%-ot (Oravecz és Dienes 2002).

A feldolgozott szövegeket a *Heti Világgazdaságból* merítettük. A választás szándékosan azért esett erre a folyóiratra, mert benyomásunk szerint a cikkek olyan kimunkált, időnként már-már mesterkélt stílusban íródtak, amelyek nagy számban tartalmaznak rendkívül összetett NP szerkezeteket. Bízvást állíthatjuk tehát, hogy ez a szöveg igazán próbára teszi az annotáló rendszert. Ugyanakkor azonban ezt a tényt érdemes figyelembe venni az eredmények értékelésekor.

4 A fejlesztő eszköz

Az NP annotálási szabályok fejlesztését a CLaRK rendszer (Simov et al. 2002) segítségével végezzük. A CLaRK rendszer egy XML alapú korpuszfeldolgozó eszköz, amely három technológia egyesítésével biztosítja a hatékony szövegannotációt: az Xpath mechanizmus biztosítja a szöveg tetszőleges részének elérését, a beépített véges automata dolgozza fel a reguláris kifejezésekkel definiált nyelvtant, és az ú.n. megszorítás (constraint) szabályok alkalmazásával növelhetjük az XML technológia rugalmasságát.

A legalsó szinten egy tokenizáló modul bontja fel a szöveget a kívánt egységekre. A tokenizáló szabályok tetszés szerint definiálhatók, lépcsősen egymásra épülnek, és akár minden szabályhoz külön-külön is hozzárendelhetők. A szöveg feldolgozásának központi eleme a lépcsős reguláris grammatika, amelynek szabályaihoz az Xpath kifejezések segítségével definiáljuk a szabályok hatókörét és a szöveg feldolgozandó elemeit. A nyelvtani szabályok meghatározásakor módunk van a reguláris kifejezések bal és jobb oldalán lévő szövegkontextus definiálására. A szabályok kimenete egy XML annotáció, amelyet általában arra használunk, hogy a szabályra illeszkedő szövegrész köré XML kódokat ültessünk. A nyelvtan lépcsős jellegét az biztosítja, hogy az egyes szabályok kimeneteként előállt egységek szerepelhetnek a későbbi szabályok bemenetében. Az XML annotáció jól illeszkedett a nyelvtan hierarchikus szerkezetéhez és az Xpath kifejezések valamint a constraint szabályok alkalmazásával könnyen meg lehetett fogalmazni olyan szabályokat, mint például a head jegyeinek perkolációját a legfelsőbb kiterjesztési szintre még akkor is, amikor az összetett NP struktúra miatt a két pont igen távol esett egymástól.

5. Az NP annotáció általános elvei

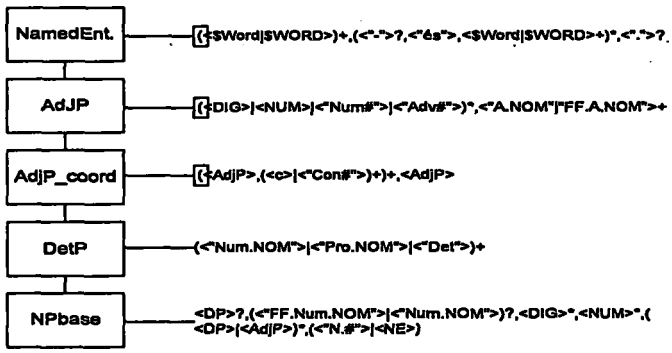
A 2. részben ismertetett sajátosságokat figyelembe véve a következő elvekre építettük a főnévi felismerő szabályainkat. Mivel a magyarban a főnévi csoport belső szerkezete balra rekurzív, az NP bal szélső eleme az NP feje, amit alapfeltevésként azaz a szabályok első körében egy N tölt be. A leghosszabb illeszkedő mintát használtuk a reguláris kifejezésekben. Az NP-n belül szerepelhet módosító szerepben N is, de csak nominatívusz esetben. A teljes NP annotációs nyelvtan két szakaszra bomlik: az elsőben meghatározzuk azokat az egyszerű NP-ket, amelyeknek a feje N vagy tulajdonnév

(NamedEntity). Amint az az 1 . ábrán látható, ez a szakasz is lépcsősen egymásra hivatkozó szabályokból épül fel.

Arra való tekintettel, hogy a magyarban a főnévi csoportok fejének szerepét a főnevek hiányában módosító elemek is átvehetik, az elemzés további szakaszában a „depth-first" stratégiát követtük, vagyis először az N fejű összetett főnévi csoportok szerkezetét határoztuk meg egészen addig, amíg a szabályok már nem találtak illeszkedő adatokat, majd ezután következett a nem N-fejű NP-k feldolgozása. Még itt is két szakaszt kellett elválasztanunk, először ugyanis csak olyan NP-ket határoztunk meg, melyekben a fej szerepét nem igenév tölti be, majd csak ezek kimerítő lefedése után engedhettük meg az igeneveket fej szerepben (ld. NP2 és NP3 a 2. ábrában). Külön problémát jelentett az igeneves szerkezetek előtt álló módosító elemek jobb szélének a meghatározása. Jobb híján kénytelenek voltunk megengedni tetszőleges NP bővítményt, ami kétségtelenül a túlgenerálás egyik forrása lehet.

6. Az NP felismerő szabályrendszer

A kidolgozott szabályokat az 1. és a 2. ábra tartalmazza. Amint látható a szabályok egyaránt hivatkoznak szintaktikai osztályokra (<DP>), msd attributumokra (<"FF.Num.NOM">) és szóalakokra (<"és">). A reguláris kifejezések sajátos notációjának leírását a CLaRK rendszer leírásában találhatjuk (Simov 2001).



1. ábra. Az alapszintű NP-k szabályrendszere

7. Eredmények

A szabályrendszert 100 kézzel azonosított mondaton (gold standard) teszteltük. A tesztelésben a legfelsőbb szintű NP-k helyességét vizsgáltuk. A 2537 tokent tartalmazó tesztszöveg összesen 488 mondatszintű NP-t tartalmazott. Két mérőszámot is célszerűnek tartottunk alkalmazni, az egyik a szerkezetekre vonatkoztatva mutatja a pontosság és lefedettség számait, a másik az érintett szövegszavakban méri ugyanezt.

(i) szerkezeti mutatószámok:

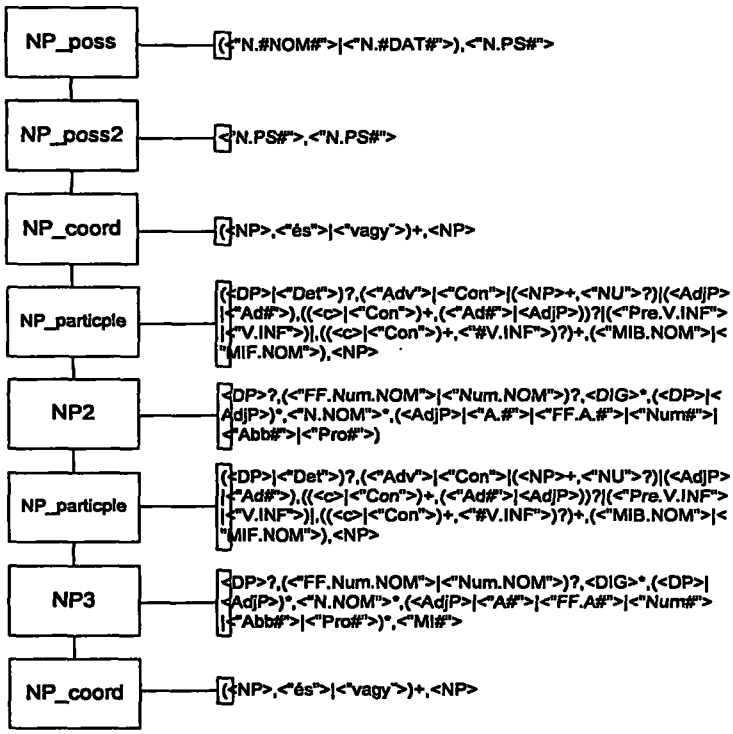
- pontosság: a kézzel ellenőrzött és a mintában egyaránt szereplő NP-k száma/ a mintában szereplő NP-k száma

- lefedés: a kézzel ellenőrzött és a mintában egyaránt szereplő NP-k száma / a kézzel ellenőrzött anyagban szereplő NP-k száma

(ii) szóalak mutatószám : Ugyanaz a két arány, mint (i)-ben, de nem NP-k ben, hanem a szóalakok számában meghatározva.

Az FB1 értékeket a szokásos módon, az alábbi képlet szerint számoltuk: FB1=2*pontosság * lefedés/(pontosság + lefedés).

Az eredményeket az 1. és a 2. táblázatban foglaltuk össze.



2. ábra. Az összetett NP szerkezeteket előállító szabályrendszer

A számszerű mutatókat az érintett szavak tekintetében elfogadhatónak tekinthetjük. A nyelvtan kimenetét minőségileg vizsgálva a benyomásaink kedvezőbbek annál, mint amit a számok tükröznek. Az eltérést részben az indokolja, hogy, amint azt a 3. részben említettük, a feldolgozott szöveg a főnévi csoportok szempontjából több tekintetben is extrémnek tekinthető. A szabályrendszer kimenetének hibaelemzése jelenleg is folyik. A további munka kereteit egyértelműen kijelölik a jelenlegi szabályok által lefedett jelenségek ismert korlátai.

NP szám gold standard-ban:	488
NP szám a mintában:	611
Helyes NP-k száma	323
NP pontosság:	52.87%
NP lefedettség:	66.17%
FB1:	58.78%

1. táblázat NP szerkezeti mutatószámok

Szószám a gold standard-ban:	1660
Szószám a mintában::	1577
Szószám a helyes NP-kben:	1511
Szószám pontosság:	95.81%
Szószám lefedettség:	91.02%
FB1:	93.36%

2. táblázat Szóalak szerinti mutatószámok

Hivatkozások

Abney S 1996 Partial Parsing via Finite-State Cascades In *Proceedings of the ESSLLI'96 Robust Parsing Workshop,* pp 1 – 8

É. Kiss K 1994 Sentence structure and word order. In: Kiefer–É. Kiss (eds): *The Syntactic Structure of Hungarian.* San Diego, Academic Press. 1–90.

Oravecz Cs, Dienes P 2002: Efficient stochastic part of speech tagging for Hungarian. In *Proceedings of the Second International Conference on Language Resources and Evaluation,* Las Palmas, pp 710—717

Prószéky G, Tihanyi L 1996 "Humor -- a Morphological System for Corpus Analysis." *Proceedings of the first TELRI Seminar in Tihany.* Budapest, pp 149-58.

Simov K 2001 CLaRK – an XML-based System for Corpora Development in *Proceedings of the Corpus Linguistics 2001 Conference,* Lancaster, pp 553-560.

Simov K et al. 2002 CLaRK System: Construction of Treebanks in *The First Workshop on Treebanks and Linguistics* Theories Sopozol: LML CLPP Bulgarian Academy of Sciences 183-199.

NP annotation using the CLaRK system

Tamás Váradi

Linguistics Institute,
Hungarian Academy of Sciences
1068 Budapest, Benczúr u 33
varadi@nytud.hu

Keywords: partial parsing, NP annotation, cascaded regular grammars

The paper presents interim results of work in-progress to develop a robust NP annotation system based on finite state technology. The grammar uses the notions of cascaded regular grammar proposed by Abney (1995). The input text selected for the analysis was taken from *Heti Világgazdaság* on account of its extremely elaborate style containing numerous very complex NP's. The text was processed with the same technology developed for the Hungarian National Corpus as a result of which each word has its lemma and morphosyntactic description stored with it. The disambiguation process developed by Oravecz and Dienes (2002) was around 98 %.

Hungarian has some peruliarities which bars the straightforward adaptation of parsing techniques developed for other languages. Its word order, better to say, order of constituents is relatively free, while word order within constituents is bound. The dificulties making the automatic recognition of NP boundaries include the possible replacement of its head with its modifiers and the left recursive insertion of participles (progressive and perfect), which can bring in an unspecifiable and open-ended list of their modifiers.

The grammar was developed with the CLaRK system (Simov 2001), an XML based corpus processing software tool containing a finite state grammar compiler and a variety of other technologies, which altogether make this environment highly suitable for text annotation. The NP annotation rules rely heavily on the cascaded use of regular expressions defining increasingly complex NP's in two main stages. First, base NP structures containing noun heads are created. In the second stage more complex NP's produced through coordination and merge of possessive NP's are defined including those that have heads other than nouns.

The results of the analysis are tested against a hand-compiled corpus of a hundred sentences. Precision and recall figures are given both in terms of number of NP's and number of tokens involved. The numerical per-token recall and precision measure is quite acceptable and an intuitive evaluation of the parse output gives a better impression, considering the extremely elaborate NP structures that are successfully analyzed.

Főnévi csoportok tanulása és felismerése

Hócza András, Iván Szabolcs

Szegedi Tudományegyetem, Informatika Tanszék
6720 Szeged, Árpád tér 2.
hocza@inf.u-szeged.hu, ajven@programozo.hu
http://www.inf.u-szeged.hu

Kulcsszavak: főnévi csoportok felismerése, szabály alapú módszerek

Kivonat. A dolgozat azt tanulmányozza, hogy főnévi szerkezetek felismerése milyen részproblémákra bontható, illetve, hogy az egyes részproblémákban, milyen elemzések, teszteredmények segítenek bennünket a továbblépésben a lehető legjobb minőségű megoldás felé. A számos megközelítési lehetőség közül mi a szabály alapú módszereket választottuk, de ez is felvet számos specifikus részproblémát. Két tanuló algoritmust alkalmaztunk szabályok előállítására. Az egyik a közismert *C4.5*, a másik egy saját fejlesztésű algoritmus, az *RGLearn*. A teszteket egy erre a célra kifejlesztett NP elemzővel végeztük.
A kísérleteket és a különféle teszteket jelentős mértékben segítette a körülbelül 1,2 millió szót tartalmazó, kézzel annotált Szeged Korpusz [1], amely különböző (iskolai, szépirodalomi, számítógépes, jogi, üzleti) szövegtípusokra tartalmazza a nyelvészeti szakértők által bejelölt főnévi csoportokat.
Az NP felismerésre kifejlesztett elemzőnk, szakértői szabályokkal 65%-os, környezetfüggetlen szabályokkal 85%-os, környezetfüggő szabályokkal 90%-os pontossággal építette fel tesztállományban található NP szerkezeteket.

1 Bevezetés

A főnévi csoportok (továbbiakban: NP – Noun Phrase) tanulása rendkívül összetett probléma. Amikor egy ilyen problémára gépi tanulási technika segítségével szeretnénk megoldást találni, számos alternatíva nyílik meg előttünk. A választási lehetőségekre hozott döntések jelentős mértékben befolyásolhatják végeredmény minőségét.

Az NP felismerés az automatikus információ kinyerést [2], [3], [4], [5] megvalósító programlánc (ToolChain) részmodulja. Ez azt jelenti, hogy a felismerés minősége függ a megelőző modulok minőségétől is. Más szempontból viszont ez lehetőséget ad olyan megoldásokra, hogy bizonytalan esetekben döntést a folyamat következő moduljaira hárítsuk.

1. Ábra: Az NP felismerés helye az információ kinyerés folyamatában.

Munkánkat jelentős mértékben támogatta az elkészült álló Szeged Korpusz [1] amely különböző (iskolai, szépirodalomi, számítógépes, jogi, üzleti) szövegtípusokra tartalmazza a nyelvész szakértők által bejelölt főnévi csoportokat. A gépi tanuláshoz szükséges tréning és teszt adatokat a korpuszból vettük. A tesztelés során a kiértéke-

lést, a manuális bejelölés minőségi ellenőrzésénél is jól bevált saját fejlesztésű (NpCheck) algoritmussal végeztük, amely összehasonlította az előállított és etalon XML fájlokat, hogy milyen mértékben térnek el a bennük tárolt NP szerkezek.

A következő fejezetekben az általunk alkalmazott szabály alapú módszer kialakításának részleteit ismertetjük.

2 Az NP felismeréshez alkalmazott módszer kiválasztása

Számos megközelítési lehetőség van az NP felismerés kapcsán. Igen népszerűek és hatékonyak a különféle valószínűségi modelleket alkalmazó módszerek, mint például a HMM [5]. Ezeknél a módszereknél a döntést támogató modell egy valószínűségi értékeket tartalmazó számhalmazzal van reprezentálva, melyek emberi szem számára nem értelmezhetők, módosítani, belőlük következtetéseket levonni nem lehet. A szabály alapú módszereknek viszont számos előnyük van:

- Könnyen áttekinthetők, értelmezhetők.
- Könnyen kiegészíthetők szakértői tudás beépítésével, amelyek megnyilvánulhatnak szakértők által adott szabályokban, kezdeti hipotézisben, vagy a meglévő szabályok finomításában
- A támogatás gyors és egyszerűen megvalósítható.
- A szövegben rejlő mintázatok jók kereshetők és reprezentálhatók.

A szabály alapú módszerek is számos alternatívát vetettek fel. Alapvetően a megvalósítandó rendszer döntéstámogatáshoz használhat szakértői szabályokat, vagy valamilyen gépi tanulási technika által előállított szabályokat.

2.1 Szakértői szabályok

A Szeged Korpusz létrehozásának a NP bejelölési fázisában ki lett dolgozva egy útmutató a manuális munkát elvégző nyelvész szakértők számára, annak érdekében, hogy a körülbelül 15-20 fő által elvégzendő munka egységes irányelvek szerint legyen megvalósítva. Azonban ez az útmutató számos helyen tartalmazott intuitív elemeket, mely információ a számítógép nyelvére nehezen fordítható le. Történtek kísérletek nyelvészeti szakértők által megadott szabályrendszer kialakítására, de a fentebb említett problémák miatt a szabályrendszerbe végül csak a legbiztosabb (95%-100% találati pontosságú) szabályok kerültek be. Ez azt jelentette, hogy ha az elemző döntött valamilyen szabály alapján, az nagy valószínűséggel jó volt, de nagyon sok NP-t nem jelölt be. A szakértői szabályok végül a manuális munka felgyorsításában játszottak szerepet, a CLaRK[1] rendszerrel végzett előfeldolgozás segítségével. A szakértői szabályok felhasználásával az NP elemző átlagosan 65%-os pontosságot ért el.

2.2 Szabály alapú gépi tanulási technikák alkalmazása

A Szeged Korpusz feldolgozásának különféle szakaszban alkalmaztunk ILP (*Inductive Logic Programming*) módszereket különféle szabályrendszerek előállításánál. Az IMPUT algoritmus [6] a szabályok specializálásával képes megjavítani egy

[1] Programfejlesztő: Kiril Simov, BulTreeBank Project, Linguistic Modelling Laboratory, CLPP, Bulgarian Academy of Sciences (http://www.bultreebank.org).

szabályrendszer pontosságát, azonban kellett ehhez szükség volt egy szabályrendszert előállító módszerre is. Másrészt a Prológ rendszer lehetőségeit nehéz lett volna átvinni egy önállóan működni képes programba, márpedig a automatikus információ kinyerés részeként nekünk ilyen modulra volt szükségünk. Így a C4.5 [7] döntési fa készítő algoritmus és egy saját fejlesztésű módszer az RGLearn (*Rule Generalization Learner*) mellett döntöttünk végül.

3 A tanulási példák előállítása

A tanulási fázis megkezdése előtt az XML fájlokban tárolt információkat át kell alakítani gépi tanulásra alkalmas formába, azaz a rendelkezésre álló információkat úgy kell átszervezni, hogy az tanulási probléma legyen. Szabály alapú tanulás esetén ez egy táblázattal reprezentálható, melynek a sorai a ***tanulási példák***, az oszlopai az ***attribútumok*** és a táblázat egy kitüntetett oszlopa a ***döntés***. A tanulás lényegében a táblázatnak a döntés szerint ellentmondásmentes tömörítéséből (általánosításából) áll.

2.2 Főnévi szerkezetek előfeldolgozása

A tanulás minőségét több előfeldolgozással kapcsolatos tényező is befolyásolhatja:

- **Az attribútumok száma:** Ha túl sok az attribútum, az jelentős mértékben lelassíthatja a tanuló futását.
- **Az attribútumok információ tartalma:** Ha nincs elég információ, a tanuló nem tud elfogadható pontosságú szabályokat előállítani.
- **A példák száma:** Minél több a példa, annál jobb az eredmény.
- **Redundancia:** Ha egy adott típusból sok hasonló példa van, más típusból viszont kevés az ronthatja az eredményt.
- **Konzisztencia:** Ha a példák sok hibát, ellentmondást tartalmaznak az szintén rontja az eredményt.

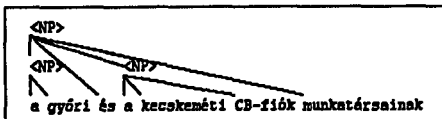
Ezért az előfeldolgozásnak jelentős szerepe van abban, hogy milyen minőségű lesz a gépi tanulással előállított szabályrendszer, tehát ez a lépés nem egyszerűen csak konvertálás, hanem valóban a tanulási probléma megszerkesztése.

2.2 Főnévi szerkezetek előfeldolgozása

A főnévi szerkezetek többnyire fa struktúrát alkotnak, ez azt jelenti, hogy egy NP magasabb szinten összekapcsolódhat egy vagy több másik NP-vel vagy szóval, együtt egy újabb NP-t alkotva. A gépi tanulás megvalósításához a fa struktúrát le kell bontani elemi fa építő utasításokká. A tanulási példák szókörnyezet, döntés párokból állnak, a tanulási probléma az lesz, hogy egy adott szó pozíciója kezdő eleme-e egy NP-nek

- **Szókörnyezet:** szavak és MSD kódok a vizsgált szó pozíciójából kiindulva. A következő attribútumok vannak: ..., CL2, CL1, C, CR1, CR2,...,WL2, WL1, W, WR1, WR2,... (pl: CL3 attribútum jelentése: a vizsgált szótól balra 3-mal lévő szó MSD kódja.)
- **Döntés:** Lehetséges értékei: NONE, NP1, NP2, NP3,... (pl: NP3 szimbólum jelentése az, hogy a vizsgált szó kezdő szava egy 3 szó hosszú NP-nek).

A főnévi szerkezetek lebontása több menetben történik, addig tart amíg van feldolgozható NP. Mindig a legbelső NP kerül először lebontásra.




```
<NP>
     <NP>
          a [Tf]
          győri [Afp-sn]
     </NP>
     és [Ccsw]
     <NP>
          a [Tf]
          kecskeméti [Afp-sn]
          CB-fiók [Nc-sn]
     </NP>
     munkatársainak [Nc-pg---s3]
</NP>
```

Példák kiírása:
[az 'a' szó környezete], NP2
[a 'győri' szó környezete], NONE
[az 'és' szó környezete], NONE
[az 'a' szó környezete], NP3
[a 'kecskeméti' szó környezete], NONE
[a 'CB-fiók' szó környezete], NONE
[a 'munkatársainak' szó körny.], NONE

A legbelső NP-k helyettesítése:

```
<NP>
     NP
     és [Ccsw]
     NP
     munkatársainak [Nc-pg---s3]
</NP>
```

2. Ábra: Egy főnévi szerkezet lebontásának és a példák kiíratásának menete.

3 A tanuló algoritmus

A C4.5 algoritmus az ID3-algoritmus egy továbbfejlesztett változata. J. R. Quinlan[2] nevéhez fűződik. A C4.5 egy döntési fát állít elő, melyben a csomópontok egy-egy attribútumra vonatkozó kérdések, a levelek pedig a döntések. A C4.5 úgy próbálja előállítani a döntési fát, hogy minél kevesebb kérdéssel el lehessen jutni a döntéshez, ezért azokat az attribútumokat választja ki csomópontoknak, melyeknek legnagyobb az információs nyeresége. A döntési fa pedig átkonvertálható szabályokká.

Azonban szükség volt egy olyan algoritmusra, amely megengedi, hogy az NP felismerés specifikus részproblémáit is implementálni lehessen. Ezt a C4.5 nem támogatta. Ezért fejlesztettük ki az RGLearn algoritmust, mely egy kezdeti szabályrendszerből kiindulva úgy próbálja azt általánosítani, hogy az általánosítással bejövő hiba egy előre megadott küszöbérték alatt maradjon. A kezdeti szabályrendszer lehet szakértők által megadott szabályok, vagy a tréning példákból generált, nem alapértelmezett döntést tartalmazó esetek (NP tanulás esetén ezek a NP1, NP2, ... döntés esetei).

```
RULE_SET = non default cases from EXAMPLE_SET
while change RULE_SET do
{
     foreach RULE of RULE_SET do unification RULE
     foreach RULE of RULE_SET do generalization RULE
     foreach RULE of RULE_SET do delete rules covered by RULE
}
```

Unification RULE:
Összevonja RULE szabályt azzal a szabállyal ami hozzá legjobban hasonlít, ha az így kapott szabály pontossága nagyobb egy előre megadott küszöbértéknél.

Generalization RULE:
Általánosítja a RULE szabályt úgy, hogy egy attribútumot értékét általánosabb reguláris kifejezésre cseréli, vagy elhagyja, ha az így kapott szabály pontossága nagyobb egy előre megadott küszöbértéknél.

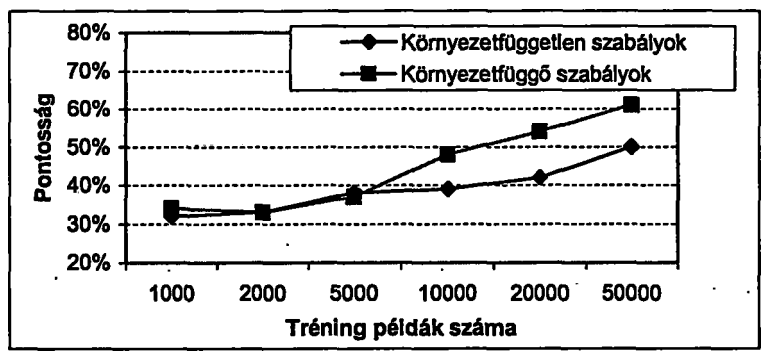
[2] J. R. Quinlan: C 4.5: Programs for Machine Learning, Morgan Kaufmann Publishing, (1993).

4 NP felismerés

Az NP felismerésnél az előfeldolgozás fordítottja, az NP szerkezetek felépítése történik a legbelső NP-ből kiindulva. Akkor jó az NP elemző, ha a tréning példákon nagy pontossággal reprodukálni tudja a megtanult szerkezeteket és ismeretlen teszt szövegen is jó hatásfokkal, az etalonnal egyezőnek ismer fel. A felismerés módja azonban számos kérdést felvet.

4.1 Környezetfüggő vagy környezetfüggetlen szabályok?

Az alábbi grafikonon található mérési eredmények arról informálnak bennünket, hogy az egyforma anyagon, gépi tanulással előállított környezetfüggő szabályok a tréning példák növelésével dominánsan jobbak, mint a környezetfüggetlen szabályok. Ez a különbség a többletinformációból adódik. A teljes korpuszon végzett gépi tanulással kapott szabályrendszer környezetfüggetlen esetben átlagosan 85%, környezetfüggő esetben átlagosan 90%-os eredményt ért el a tesztállományokon.



3. Ábra: A környezetfüggő és környezetfüggetlen szabályokkal végzett NP felismerés eredményei különböző számú tréning példán

4.2 Mohó algoritmus vagy a döntés elhalasztása?

Az NP elemző által használt szabályrendszer lehetővé teszi a szabályok közötti gyors (bináris) keresést. Az először a legjobb szabályokat (melyek kevés hibával sok esetet lefednek) próbálja meg alkalmazni. A végrehajtás két lépésben történik: először van a lehetséges szabályalkalmazások bejelölése a teljes mondatra. A második lépésben történik a bejelölések végrehajtása. Vitás esetben, ha a bejelölt NP-k átlógnak egymásba, a legjobb szabály által bejelölt NP mellett dönt az elemző. A végrehajtás addig megy, amíg van bejelölhető NP. A NP elemző algoritmusa a következő:

```
while change do
{
    foreach tag of sentence do
    foreach rule of ruleset do
    if rule covers tag then sign tag with rule

    foreach tag of sentence do
    if tag sign with better rule then subtitute NP
}
```

A fentebb ismertetett (*mohó*) algoritmusnak az az előnye, hogy egyből előállít egy nagy valószínűséggel jó NP struktúrát egy adott mondatra. Azonban van számos hátránya is, például nem biztos, hogy a legjobbat. Az például jobb megoldás lenne, ha előállítaná az összes lehetséges szerkezetet, úgy, hogy valamilyen heurisztikát használva minden lehetőséghez egy valószínűségi értéket rendelne, a következő szempontok szerint:

- Az alkalmazott szabályok összesített valószínűsége.
- Az NP szerkezetek komplexitása (szintek, levelek száma) minél nagyobb.
- Csak olyan megoldások fogadhatók el melyek lefedik az összes főnevet.
- A legjobb megoldások közül egy későbbi fázis (ontológia, szemantikus keretek illesztése) plusz információi alapján választunk.

Ez utóbbi módszer még fejlesztés alatt van, de a Morphologic Kft által kifejlesztett HumorEsk program felhasználásával lesz megvalósítva.

5 Összefoglalás és fejlesztési lehetőségek

A dolgozatban bemutattuk egy szabály alapú NP tanuló és elemző rendszer előállításának lépéseit. Vázoltuk azt, hogy egyes lépésekben milyen lehetséges megoldások közül választhattunk és milyen megfontolások és előzetes teszteredmények alapján választottuk ki az általunk ítélt legjobbat.

A közeljövőben szeretnék még több módszert kipróbálni, összehasonlítani, hogy még hatékonyabb megoldást kapjunk az információ kinyerésnek erre az igen fontos részproblémájára. Az egyik lehetőség, hogy megvalósítva az összes NP szerkezet előállítását komplexebb problémaként próbálunk választani, esetleg bevonva későbbi fázisokat is, mint ontológia, szemantikus keretek illesztése. A másik lehetséges irány, hogy a szabály alapú módszerek mellett más, például HMM módszert vagy esetleg több módszer kombinációját alkalmazzuk a problémára.

Irodalom

1. Alexin, Z., Csirik, J., Gyimóthy, T., Bibok, K., Hatvani, Cs., Prószéky, G., Tihanyi, L. (2003) *Manually Annotated Hungarian Corpus*, in Proceedings of the Research Note Sessions of the 10th Conference of the European Chapter of the Association for Computational Linguistics EACL03, Budapest, Hungary, pp. 53–56.
2. Hócza, A., Alexin, Z., Csendes, D., Csirik, J., Gyimóthy, T.: *Application of ILP methods in different natural language processing phases for information extraction from Hungarian texts* in Proc. of the Kalmár Workshop on Logic and Computer Science, Szeged, Hungary, 1-2 October, pp. 107-116 (2003)
3. Freitag D. (2000) *Machine Learning for Information Extraction in Informal Domains*, Machine Learning, 39, 169–202.
4. Califf, M. E. and Mooney, R. J. (2001) *Bottom-Up Relational Learning of Pattern Matching Rules for Information Extraction*, in Journal of Machine Learning Research
5. Freitag and McCallum (2000) *Information extraction with HMM structures learned by stochastic optimization*, in Proceedings of the Seventeenth National Conference on Artificial Intelligence, 2000.
6. Alexin, Z., Gyimóthy, T., and Boström, H. (1997) *IMPUT: An Interactive Learning Tool based on Program Specialization*, Intelligent Data Analysis (IDA) Journal, Vol 1 No 4, Elsevier Holland.
7. Quinlan, J. R. (1993) *C 4.5: Programs for Machine Learning*, Morgan Kaufmann Publisher.

Learning and recognizing noun phrases

András Hócza, Szabolcs Iván

University Of Szeged, Department of Artificial Intelligence
hocza@inf.u-szeged.hu, ajven@programozo.hu

Keywords: noun phrase recognition, rule based methods

Abstract Learning noun phrases is a very complex problem, therefore it can be divided into several sub-tasks. In the present paper, authors try to examine the type of sub-tasks there are and the way they can be solved in order to achieve the final aim: an efficient noun phrase recognition tool. Several different approaches exist, out of which we have chosen rule based methods due to some advantages they have over other approaches (e.g. rules are easily understood not only by programmers but also by linguistic experts; rules can be extended with expert knowledge). In our work, we used two different rule based learners: the first one is the well-known *C4.5* algorithm, the second one is the so-called *RGLearn*, developed by the authors of the present paper. *RGLearn* proved to have some advantages over *C4.5*, because it is simpler to build problem specific parts into it. As a result of the learning process, the learners produced context free and context dependent rule sets. The preprocessing step is a very important part of the learning procedure, where we have to define what the learning problem exactly is. We have to make sure that the method will really learn what we want it to learn, that the given information is enough for the learner, and that the conversion creates consistent training examples without redundancy.

In this paper, we demonstrate how manually annotated sentences can be transformed into learning problems. In the first step, we dismantle noun phrase structures into elementary tree building commands. Then we generate training examples from every word position and based on current context decide whether we have to use a tree building command.

Noun phrase recognition is done by a greedy, bottom-up algorithm, that builds up the noun phrase structure of a sentence. We compared the results of the automated noun phrase recognition with manually annotated example sentences. The comparison was performed on the Szeged Corpus, which is a manually annotated textual database containing approx. 1.2 million words. The context dependent rule set was found to be the best with 90% per word accuracy, then came the context free rules with 85%, and finally expert's rules only performed 65%. Considering precision, expert rules provided rather good results (95%-100%), therefore we chose them to preprocess a number noun phrases before manual annotation in order to help the work of the annotators.

Noun phrase recognition is an important part of *Information Extraction.* The aim of our research group on the long run is to develop a modular *ToolChain* for information extraction where one of the modules will be the described noun phrase recognizer. Here we have to note that some errors may come from previous phases of automated analysis conducted by the modules of the *ToolChain,* which can cause errors in noun phrase recognition as well. At the same time, the *ToolChain* provides the possibility to solve the problem of noun phrase recognition in another way: if the parser generates the more possible noun phrase structures, the following modules of the *ToolChain* (ontology, semantic frame recognition) can select the best one by using extra information.

GeLexi projekt: GEneratív LEXIkonon alapuló mondatelemzés

Alberti Gábor, Kleiber Judit, Viszket Anita

Pécsi Tudományegyetem, BTK, Nyelvtudományi Tanszék
H7624 Pécs, Ifjúság útja 6., Magyarország
gelexi@btk.pte.hu
http://lingua.btk.pte.hu/gelexi.asp

Kivonat. A 2001-ben Pécsett alakult *GeLexi* kutatócsoportunk[1] kiinduló célja a számítógépes implementálás révén való legitimálása egy olyan grammatikának, amely a generatív nyelvészeti kutatásokban a hatvanas évektől egyre erősödő *lexikalista* tendenciát a „végsőkig" fokozza: lemond a frázisstruktúra-építésről, miközben a szórendről mégis számot ad, rangsorolt szomszédossági követelmények segítségével. Másik újdonsága a „totális lexikalizmus" kiterjesztése a morfológiára is: nem a szavakhoz, hanem a morfémákhoz tartoznak a minden grammatikai szintről egyidejűleg információt hordozó lexikai egységek, amelyek oly módon „szerelik össze magukat" a mondatelemzés során, hogy az is eldől, mely elemek épülnek össze szóvá, és melyek alkotnak (egymás „közelében" maradó) külön szavakat. Nyelvelméleti törekvéseink gyakorlati hozadéka egy folyamatos fejlesztés alatt álló program, amely egy (jelenleg magyar vagy angol) szósoron elvégzi a „generatív grammatikai alapfeladatokat": eldönti, hogy jól formált szavakból álló grammatikus mondattal áll-e szemben, majd (a pozitív esetben) kétféle diskurzus-szemantikai elemzést kínál. E cikkben *és* kötőszavas mondatok elemzése szemlélteti majd eljárásainkat.

1 Bevezetés

Általános célunk annak igazolása, hogy a költség–haszon elvet mindenek fölébe helyező, „sekélyelemző" szemléleten érdemes lenne túllépnie a számítógépes nyelvészetnek, visszafordulva a tiszta nyelvelméleti alapok felé. Kidolgozható ugyanis olyan formális (generatív) grammatika [4], amely éppen a modern számítástechnikában előnyösnek mondott kapacitásmegosztást mutatja: „minimális processzálás – maximális adattár".[2]

[1] A cikk megírását és a szegedi konferencián való jelenlétünket a T 38386 számú OTKA pályázat, valamint az első szerző esetében a MTA Nyelvtudományi Intézetének Hajdú Péter Vendégkutatói Ösztöndíja tette lehetővé. Köszönettel tartozunk továbbá Balogh Katának, aki a jelenlegi mondatelemző szoftverünk szintaktikai és szemantikai részeinek a nagy részét írta, de sajnos már nem tagja a GeLexinek, mivel (szerencsére) amszterdami doktoranduszhallgatóként intenzív szemantikaelméleti kutatásokba fogott.

[2] Összahangban azzal a lehetőséggel, amelyről Prószéky (persze a MorphoLogic gépi fordítási projektuma kapcsán) így ír: „a memóriakapacitás korábban nem tette lehetővé ilyen számú és méretű minta egyidejű használatát". Közben a kezdetekben [15] maximálisan „processzálás-párti"

Kutatócsoportunk kiinduló célja (ld. a 2. pontot) az előbbiekben közölt céllal nagy átfedést mutat, lényegében annak az elméleti nyelvész szemszögéből való megfogalmazásáról van szó: a számítógépes implementálás révén kívántunk legitimálni egy olyan (homogén felépítéséből adódóan metaelméleti érdekességgel is bíró [4]) grammatikát (GASG: *Generatív argumentumszerkezet-grammatika*), amely az iménti lábjegyzetben említett *lexikalista* tendenciát a „végsőkig" fokozza: lemond a frázisstruktúra-építésről, miközben a szórendről mégis számot ad, rangsorolt szomszédossági követelmények segítségével (egyesítve az eddigiekben meghivatkozott frázisstruktúra-nyelvtanok előnyeit a függőségi nyelvtanokéival [23]).

A 3. pontban a kutatócsoport hároméves tevékenységét tekintjük át, azzal a feltételezéssel élve, hogy ez az első magyar számítógépes nyelvészeti konferencia elsősorban a műhelyek bemutat(koz)ására szolgál.[3]

A 4. pontban felvillantjuk mondatelemző Prolog-programunk jelenlegi tudását, két futtatás végeredményéből idézve, különös tekintettel az *és* kötőszóra (annak a stratégiánknak a jegyében, amelynek értelmében minden koferencián bemutatunk valamilyen új eredményt túl az általános koncepció összefoglalásán).

2 A kiinduló cél

A *generatív* nyelvészetnek [15] elévülhetetlen érdeme, hogy *formális elméletet* nyújtott annak a régi felismerésnek a megragadására, miszerint a mondatjelentésnek két forrása van: a lexikai elemek és az azokat összefűző szerkezet. Kezdetben e szerkezetek kombinációs lehetőségeinek a matematikai vizsgálata jelentette azt a központi kérdést —ld. a Chomsky-féle nyelv(tan)osztályok témakörét, különös tekintettel a környezetfüggetlen és környezetfüggő grammatikákra [15, 21]—, amelyből két „gyakorlatiasabb" tudományág is sarjadt: a generatív nyelvleírásé és a számítógépes nyelvészeté. A generatív nyelvleírásban a mondatösszetevők mozgatását megengedő Chomsky-féle irányzatnak nem sikerült elméletileg igazolnia e mozgató *transzformációk* elengedhetetlenségét [21], ezért a hetvenes évektől kezdve több olyan „eretnek" generatív irányzat is meg tudott szilárdulni [13, 14, 17, 18], amely „alig" lépte túl a környezetfüggő eszköztár kereteit (Partee-ék [21] kiválóan bemutatják ezeknek az *enyhén környezetfüggő* eszköztáraknak a variációit). Ezzel összefüggésben az új irányzatok a lexikon+szintaxis egységben a lexikonnak a korábbiaknál jóval nagyobb szerepet biztosítottak a nyelvi jellegzetességek megragadásában, a szintaxisnak általában csupán nagyon általános frázisstruktúra-építő szabályokat meghagyva. A lexikalista tendencia elsőprő erejét mutatja, hogy a kilencvenes években a Chomsky-féle „fővonal" is ugyanilyen fordulatot vett [16] a Minimalista program keretében.

A bevezetésben említett GASG a lexikalizmus totálissá fokozásának kísérletéből született, megvalósítva Karttunen „radikális lexikalizmusát" [19]: olyan nyelvtan, amelyben

(azaz szintaxis-központú) generatív nyelvészet is erőteljes *lexikalista* fordulatot vett [13, 14, 16, 18, 19]. Hadd idézzük a faépítő nyelvtanok [21] „atyjának", Joshinak (2003) két aktuális mottóját [18]: „Complicate Locally, Simplify Globally", és „Grammar ≈ Lexicon".

[3] A bemutatkozó pont beiktatása azzal a káros következménnyel járt, hogy „felborult" a hivatkozásjegyzék a sajátpublikációk javára, ami amúgy szándékaink ellen való.

nemhogy transzformációs szabályok nincsenek, de még összetevős szerkezeti fák sem épülnek, ugyanakkor a generatív nyelvészeti *alapfeladat* elvégeztetik, azaz meghatározható, levezethető a jól formált mondatok pontos halmaza, a levezetés során pedig szintaktikai és szemantikai szerkezet rendelődik az illető (jól formált) mondathoz. Egy totálisan lexikalista nyelvtanban a grammatikai „tudás" a lexikai tételek leírásába — és csakis oda — van beépítve; minden egyes szó megmondja, hogy milyen „környezeti követelményeket" kell kielégítenie egy őt tartalmazó jól formált mondatnak. Az egyes lexikai tételek természetesen nemcsak a környezeti követelmények leírását tartalmazzák, hanem a „sajátszó" jellemzését is, hiszen más szavak éppen az illető szót fogják keresni potenciális mondatokban.

A (generatív) nyelvészeti leírásokat — elvi okok miatt is! — igen fontos számítógépes implementációval igazolni, hiszen a működő algoritmusok teszik kétségtelenné, hogy jól formalizált, egzakt rendszerünk van. Az implementáló programunk tevékenysége nem más, mint az imént említett „generatív alapfeladat" végrehajtása: egyértelműen el kell döntenie egy beírt magyar szósorról, hogy az grammatikus-e, vagy sem, és amennyiben az, morfoszintaktikai és szemantikai reprezentációt kell hozzá rendelnie.

Nyelvtanunk tulajdonságairól a következő pontokban még lesz szó; most a bevezetés kezdőgondolatához szeretnénk visszakanyarodni azzal a megjegyzéssel, hogy a memóriakapacitás növekedése és a mintaillesztési eljárások fejlődése a technológia oldalán [22] szerencsésen összetalálkozhat majd a lexikalizmusnak köszönhető homogén generatív nyelvelmélettel, amit katalizálhat az intelligens alkalmazások iránt felébredő —immár nem irreális— igény. A GASG gyakorlatilag legitimálhatja a nyelvtechnológiában eddig ösztönösen alkalmazott, szókörnyezetek illesztésén alapuló heurisztikus eljárásokat, elméletileg is korrekt rendszerré továbbfejlesztve azokat.

3 A GeLexi tevékenysége

A GASG számítógépes implementációjának javaslata először 1998-ban kapott nagyobb nyilvánosságot egy debreceni nemzetközi konferencián [1], bár az első szerző az „Alkalmazott Logikai Laboratórium" (ALL) munkatársaként már a kilencvenes évek elején készített néhány belső anyagot a témakörben. A korábbi elképzelések alapján 2001-ben végre elkészült egy elemző program, amely magyar szavakból álló sorozatokról eldöntötte (pusztán a toldalékolt szavak lexikai leírására támaszkodva), hogy jól formált mondatot alkotnak-e (az adott szórend és toldalékolás mellett) [9,3].

Majd megkezdtük a GASG szemantikai reprezentációjának kidolgozását. Más nem is jöhetett szóba, mint a Kamp-féle „DRS-ek" (diskurzusreprezentációs struktúrák) [17] egy továbbfejlesztett változatának [2] implementálása, mivel tulajdonképpen a GASG létjogosultsága melletti egyik fő elméleti érvet éppen abban látjuk, hogy *kompozicionális* [21] szemantikai partnereként szolgálhat az említett ígéretes (a Montague-féle szemantikai rendszereken [21] relevánsan túllépő) diskurzus-szemantikai elméletnek. Addigi elképzeléseink alapos (főleg elméleti) bemutatásával szolgál egy elvekről, modellekről és szabályokról szóló szegedi konferencia kötete [4], amely a nyelvleírási szintek közül a morfológiát nem tárgyalja alaposabban. A megelőző időszakban ugyanis a (toldalékolt szavakhoz rendelt) lexikai egységeket egy óriási *öröklődési hálózatban* gondoltuk elrendezendőnek, elkészítésüket pedig a bevett, reguláris kapacitású eszközökre [20] kívántuk bízni.

Később azonban egyértelművé vált, hogy a GASG kívánatos teljes homogenitását az biztosítja, ha a morfológiát is „totálisan lexikalista" módon közelítjük meg: nem a szavakhoz, hanem közvetlenül a morfémákhoz rendeljük a gazdagon strukturált, minden grammatikai szintről egyidejűleg információt hordozó lexikai egységeket, amelyek oly módon „szerelik össze magukat" a mondatelemzés során, hogy az is eldől, mely elemek épülnek össze szóvá, és melyek alkotnak (egymás „közelében" maradó) külön szavakat [5, 10, 11]. Ezen a ponton teszünk említést arról a fontos technikáról, amely a GASG-ben kiváltja a —szórend meghatározásáért felelős!— frázisstruktúra-építést: olyan *rangparaméteres* szomszédossági követelményeket támasztanak a mondatba kerülő lexikai egységek egymás iránt, amelyeket *közvetett módon* is ki lehet elégíteni, magasabb rangú szomszédossági követelmények teljesítésének előresorolása révén. Összetartozó szavak közé (pl. *a lány*) így kerülhetnek még jobban „odavágyódó" szavak (*a büszke magyar lány*), hozva esetleg függelékeiket is (*a (két (jóképű)) bátyjára büszke magyar lány*). A morfotaxist hasonló rangparaméteres szomszédossági követelményekre bízhatjuk (pl. *kutyát* > *kutyáét* > *kutyámét*), azzal a szignifikáns könnyebbséggel (ami a fenti regularitási állítással korrelál [20]), hogy egy morféma nem hozhat „függelékeket" magával.

A „totálisan lexikalista morfológia" imént vázolt eszméje a nyelvtípusok közötti különbségeket egy ún. *kopredikációs hálózat* absztrakt ábrázolási szintjén [8] képes irrelevánssá tenni, hiszen mindegy, hogy a *vár-hat-l-ak* ige morfémái vagy az *I may wait for you* angol szósor szavai keresik-e egymást. Ezen a reprezentáción keresztül számítógépes *fordítást* is szeretnénk majd végezni [7], egy nem túl távoli jövőben.

A morféma forrású szemantikai reprezentációt egy mexikói konferencián mutattuk be [6], a Springer-kötet adta kapcsolódó lehetőséget pedig a GASG nyelvtantípus matematikai definíciójának publikálására használtuk fel, mintegy szabadalmaztatva ezzel.

4 A jelenlegi mondatelemzőnk

Az alábbiakban idézünk néhány sort abból a hatvanhatból, amit programunk az 1. sorban látható szósorról mint egy `gramm("Mari...")` . *célfüggvény* tartalmáról közöl.

Kezdjük a végén: a 28. sorban *grammatikusnak* nyilvánítja a mondatot; amit nem tenne, ha nem tudta volna azonosítani a morfémáit (2. sor) megfelelő morfofonológiai környezetekben [5]. A szintaxisból két sort emeltünk ki a kötőszó kezelésével kapcsolatosan: a *Mari* főnév (`1,1`: az első szó első morfémája) a kötőszó függeléke (4.), és ez a kötőszó fogja képviselni az alanyt az igével való relációban (6.). A 9. és a 10. sor ugyanezt már a szemantikai egységek formától függetlenedett viszonyait ábrázoló *kopredikációs* reprezentáció nyelvén fejezi ki.

```
1. Mari és Juli kereshetik Pétert.
2. LEXIKAI EGYSÉGEK: ...
3. SZINTAXIS:
4. gr("noun","regent","conj",1,1,2,1)
5. ...
6. gr("noun","regent","subj",2,1,4,1)
7. KOPREDIKÁCIÓS VISZONYOK:
8. copr("and",2,1,"Mary",1,1,1,0,"arg")
9. copr("look-for",4,1,"and",2,1,1,1,"arg")
```

```
10. ...
11. SZEMANTIKA:
12. provref("fixpoint",[e(4,2,1)])
13. provref("old",[r(1,1,1)])
14. pred("Mary",1,[r(1,1,1)])
15. provref("new",[r(2,1,1)])
16. provref("<or=",[r(2,1,1),e(4,1,1)])
17. pred("element",2,[r(1,1,1),r(2,1,1)])
18. pred("element",2,[r(3,1,1),r(2,1,1)])
19. provref("old",[r(3,1,1)])
20. pred("Julie",3,[r(3,1,1)])
21. provref("new",[e(4,1,1)])
22. pred("look-for",4,[e(4,1,1),r(2,1,1),r(5,1,1)])
23. provref("new",[e(4,2,1)])
24. provref("<",[e(4,2,1),e(4,1,1)])
25. pred("may",4,[e(4,2,1),e(4,1,1)])
26. provref("old",[r(5,1,1)])
27. pred("Peter",5,[r(5,1,1)])
28. yes
```

A diskurzus-szemantikai reprezentációt teljes terjedelmében közöltük (12-27.), de a DRT-ben [17, 2] járatlanok számára csak a lényeget tudjuk közvetíteni. A 25. sor szerint *lehetséges* egy e4111 szituáció, ami nem más, mint hogy r211 *keresi* r511-et (22.), azaz az ismertnek tekintett (26.) *Pétert* (27.). Az r211 referens pedig egy r111 referensű *Mariból* (14.) és az r311-es *Juliból* (20.) álló kételemű csoport (17-18.).

A csoport alanyú igék személyragozását részben a Bánréti [12: 747. o.] által javasolt módon programoztuk le, pl. így egyeztetve: *Én és te/ti/Péter kereshetjük Marit.* Olyan *többes számú* személyragot használunk tehát, amelynek személyjegye a legmagasabb rangú koordinált személyével azonos (az imént az *én* első személye érvényesült).

Fogas kérdéseket vet fel a csoportos cselekvés interpretálása is. A jelenlegi programverzió hátterében az a szemlélet áll, hogy az egyes mondatok szemantikai ábrázolatában általában elegendő a fent illusztrált *csoportolvasat*; elegendő tehát a diskurzusba való beágyazás során később dönteni arról, hogy van-e okunk a csoport tagjainak egyedileg meghatározható szerepet tulajdonítani (pl. *disztributív* vagy *kölcsönös* viszonyt). Egyetlen esetben feltételeztünk markánsan disztributív olvasatot, egy eddig nem említett egyeztetési minta esetében: ahol egyes számban marad az ige, pl. *Péter és János keresi Marit* (és nem *keresik*, ami erősen azt sugallná, hogy együtt teszik). Vessük össze a fentivel ezt a disztributív interpretációt:

```
... pred("look-for",4,[e(4,1,1),r(1,1,1),r(5,1,1)])
    pred("look-for",4,[e(4,1,2),r(3,1,1),r(5,1,1)]) ...
```

Talán annyit e röpke szemléltetés is igazolt, hogy programunk a szokásosnál jóval szélesebb spektrumú mondatelemzést képes nyújtani egy korántsem triviális nyelvi fragmentumon. Fragmentumunk folyamatos bővítését, rendszerünk más nyelvekre való kiterjesztését és további „intelligens" nyelvtechnológiai alkalmazások kifejlesztését tervezzük; a megvalósíthatóságra a garanciát az elméletileg korrekt, gyakorlatilag pedig végsőkig egyszerűsített processzálás jelenti. Lassan kinőve a kísérleti szakaszból a Prolog helyett egy korszerű adatbáziskezelő és egy mintaillesztésben maximálisan hatékony nyelvre készülünk áttérni, és keressük a lehetőségét egy tudásbázist szimuláló korpusz beiktatásának.

Hivatkozások

1. Alberti, G.: GASG: Minimal Syntax, Maximal Lexicon and PROLOG, Paper read at ALLC/ACH '98, July 9. In Hunyadi, L. (ed.): ALLC/ACH '98. KLTE, Debrecen (1998) 81-83
2. Alberti, G.: Lifelong Discourse Representation Structures, Gothenburg Papers in Computational Linguistics 00-5 (2000) 13-20
3. Alberti, G., Balogh, K., Kleiber, J.: GeLexi Project: Prolog Implementation of a Totally Lexicalist Grammar. In de Jongh, Zeevat, Nilsenova (eds.): Proc. of the Third and Fourth Tbilisi Symp. on Language, Logic and Computation. ILLC, Amsterdam, and Univ. Tbilisi (2002)
4. Alberti, G., Balogh, K., Kleiber, J.,Viszket, A.: A totális lexikalizmus elve és a GASG nyelvtan-modell. In Maleczki, M. (ed.): A mai magyar nyelv leírásának újabb módszerei V. Szegedi Tudományegyetem (2002) 193-218
5. Alberti, G., Balogh, K., Kleiber, J.,Viszket, A.: Towards a totally lexicalist morphology. Talk at 6th International Conference on the Structure of Hungarian (ICSH6), Düsseldorf, Germany (2002). To appear in Kenesei, I., Piñón, Ch. (eds.): Approaches to Hungarian 9
6. Alberti, G., Balogh, K., Kleiber, J.,Viszket, A.: Total Lexicalism and GASGrammars: A Direct Way to Semantics. In Gelbukh, A. (ed.): Proceedings of CICLing2003 (Mexico City). LNCS N2588. Springer-Verlag, Berlin Heidelberg New York (2003) 37-48
7. Alberti, G., Balogh, K., Kleiber, J.,Viszket, A.: A fordítás totálisan lexikalista megközelítése. MANYE, Számítógépes nyelvészeti szekció, Győr (2003). Megj. előtt.
8. Alberti, G., Kleiber, J.: Extraction of Discourse-Semantic Information... In Cunningham, H., Paskaleva, E., Bontcheva, K., Angelova, G. (eds.): Information Extraction for Slavonic and Other Central and Eastern European Languages. Borovets, Bulgaria (2003) 63-69
9. Balogh, K., Kleiber, J.: Egy lexikalista nyelvtan morfoszintaxisának PROLOG-implementációja. OTDK- díjas pályamunka, JGYTF, Szeged (2001)
10. Balogh, K., Kleiber, J.: Computational Benefits of a Totally Lexicalist Grammar. In Matoušek, V., Mautner, P. (eds.): Text, Speech and Dialogue, Proceedings of TSD2003. Springer-Verlag, Berlin Heidelberg New York (2003) 114-119
11. Balogh, K., Kleiber, J.: A Morphology Driven Parser for Hungarian. Talk at the 5th Int. Tbilisi Symp. on Language, Logic and Computation. Org. by ILLC, Amsterdam, and U. Tbilisi (2003)
12. Bánréti, Z.: A mellérendelés. Kiefer, F. (szerk.) Strukturális magyar nyelvtan I. Mondattan. Akadémiai, Budapest (1992) 715-796
13. Borsley, R. D.: Modern Phrase Structure Grammar. Blackwell, Oxford Cambridge (1996)
14. Bresnan, J.: Lexical Functional Syntax. Blackwell, Oxford (2000)
15. Chomsky, N.: Syntactic Structures. The Hague, Mouton (1957)
16. Chomsky, N. (ed.): The Minimalist Program. MIT Press, Cambridge, Mass. (1995)
17. van Eijck, J., Kamp, H.: Representing discourse in context. In van Benthem, J., ter Meulen, A. (ed.): Handbook of Logic and Language. Elsevier, Amsterdam, The MIT Press, Cambridge, Mass. (1997)
18. Joshi, A. K.: Starting with Complex Primitives Pays Off. In Gelbukh, A. (ed.): Proceedings of CICLing2003 (Mexico City). LNCS N2588. Springer-Verlag (2003) 1-10
19. Karttunen, L.: Radical Lexicalism. Report No. CSLI 86-68, Stanford (1986)
20. Karttunen, L.: Computing with Realizational Morphology In Gelbukh, A. (ed.): Proceedings of CICLing2003 (Mexico City). LNCS N2588. Springer-Verlag (2003) 203-214
21. Partee, B., ter Meulen, G.B., Wall, R.P.: Mathematical Methods in Linguistics. Kluwer Academic Publ. (1990)
22. Prószéky, G.: Megértéstámogatás és gépi fordítás: nyelvtechnológia a XXI. század elején. VIII. Országos (Centenáriumi) Neumann Kongresszus (2003)
23. Schubert, K.: Metataxis (Contractive Dependency Syntax for Machine Translation). Foris, Dordrecht (1987)

GeLexi Project: Sentence Parsing Based on a GEnerative LEXIcon

Gábor Alberti, Judit Kleiber, and Anita Viszket

University of Pécs, Faculty of Humanities, Linguistics Department
H7624 Pécs, Ifjúság útja 6., Hungary
gelexi@btk.pte.hu
http://lingua.btk.pte.hu/gelexi.asp

Keywords: parsing of (Hungarian and English) sentences, Prolog, lexicalist generative grammar, DRS (discourse representation structure)

The principal aim of our Pécs research team, called *GeLexi*, is to verify that computational linguistics is worth returning from the nowadays wide-spread attitude characterized by "shallow parsing" (which is held to save expenses) to the pure theoretical (generative) linguistic basis [15, 21].

Our crucial argument relies on a double (parallel computational and linguistic) chance: to use simultaneously, on one hand, a significantly greater number of huge patterns than earlier due to the immense increase in memory capacity [22], and to work out a formal grammar, on the other hand, showing the distribution of capacity advantageous in modern computer science (in harmony with the development mentioned above): "minimal processing – maximal database". This latter chance has something to do with the sweeping *lexicalist* turn [13, 14, 16, 18, 19] in generative linguistics, which used to be chiefly "process-oriented" (i.e. syntax-centered) in its first period; the current attitude can be characterized by two mottoes of Joshi's [18], the father of *mildly context-sensitive* grammars [21]: "Complicate Locally, Simplify Globally", and "Grammar ≈ Lexicon".

What we propose is a new sort of generative grammar, *GASG* ("Generative/Generalized Argument Structure Grammar", defined in [6] and demonstrated in a wide range of papers [1-11]), which is more radically "lexicalist" [19] than any earlier one. It is a modified Unification Categorial Grammar [19, 17], from which even the principal syntactic "weapon" of CGs, Function Application, has been omitted. What has remained is *lexical sign* and the mere technique of *unification* as the engine of combining signs.

Our GASG-parser, in accordance with the basic task of every generative grammar [15, 21], decides whether a sentence is *grammatical*, and then provides a *morpho-phonological* analysis (based on a "Totally Lexicalist (approach to) Morphology" launched in [5]), a compilation of *grammatical relations*, and two kinds of *semantic* representations: a DRS [17] completed with information about its embedding in interpreters' information state also formulated as a DRS [2], and a network of *copredications*, useful in translation [7, 8].

A számítógépes szöveg négyszintű modellje

Kis Ádám
SZAK Kiadó, ELTE BTK Informatikai és Könyvtártudományi Intézet
adam.kis@szak.hu

A Neumann-elv értelmében a számítógép egységes numerikus kódot használ mind a feldolgozandó tartalomhoz, mind a feldolgozó eszközökhöz. Ez a számítógépeken leképződő világot szövegszerűvé teszi, ha a bináris kódot nyelvnek tekintjük. A hálózatok autonomizálódása révén ez a szövegszövedék az egyes számítógépek fölé kerül: ez a szöveg tárolódik, mozog és érhető el a hálózaton, és immár közömbös, milyen gépen keletkezett, és milyen továbbítja. A lényeg csupán az egységes kódolás. A digitális szöveg kétcélú. Egyrészt funkcionális, szerepe van, vezérli magát a számítógépet és azt a közeget, amelyet a számítógép hívatott vezérelni. A másik eset, amikor valóságos nyelvi szöveg a feldolgozás tárgya. Az előadás ez utóbbit kívánja felvázolni. A nyelvi szöveg 4 szinten jelenik meg: (1)A „tartalom" absztrakt formában; (2)Binárisan kódolt szöveg; (3) A megjelenő (performált) szöveg; (4)A markup nyelven megjelenő szöveg. Ezek összefüggései segítenek feltárni a számítógépekkel megvalósítható kommunikáció sajátosságait.

A számítógép és a szöveg

– „A szöveg a világ" – mondta Esterházy Péter a Mindentudás Egyetemén.
– „A szöveg a világháló alapanyaga" – mondta Varasdy Károly a Mire jó a nyelvtechnológia? című konferencián, november 4-én. [R1.]. (A cikk szövegében ritkítással, és az adott szövegrész végére írt [R*n*] jelöléssel utalok azokra a témákra, amelyeket szükséges vagy érdemes – más helyen – kifejteni.)

E két állítás között foglal helyet egy megállapítás, amely lehet akár egy elfogult textológus abszurd álma: A számítógép maga is szöveg.[R2.] Ez a kijelentés azonban csak látszatra abszurd, ugyanis, ha számítógépet nem technikailag, hanem funkcionálisan értelmezzük, olyan valóságmodellt kapunk, amelynek minden részlete leírható.

A számítógép maga virtualizálódik, fizikai mivoltában eltűnik a felhasználó szeme elől. Gondoljuk el, hogy az interneten folyó kommunikáció során igen ritkán van arra szüségünk, kapcsolati partnerünk gépéről műszaki ismereteink legyenek. Máshol sem függünk a technikától, olvasunk, képet nézünk, hangokat hallgatunk, azaz hagyományos módon használjuk az érzékeinket.

Hogy a számítógép, illetve a rajta nyugvó virtuális valóság szöveg-e, bizonyítandó, az viszont nem, hogy szöveghordozó. Ezúttal vessük meg lábunkat a biztos talajon, és foglalkozzunk a számítógépen megjelenő hagyományos szövegekkel! A szövegmegjelenítés adott formája összehasonlítható a többi manifesztációvall (beszéd, írás stb.), azaz a számítógép besorolható a szöveghordozók közé, a s z á m í t ó g é p e s s z ö v e g összehasonlítható másfajta szövegekkel, megállapíthatók h a s o n l ó s á g a i é s e l t é r é s e i .[R3.]

A számítógépes szöveg, mint nyelvi tény

A szövegtan, mint a nyelvi foglalatosságok kései ága, igen nagy hangsúlyt fektet az önmeghatározásra. Ezek a definíciókban sok az elhatárolódás, többet foglalkoznak azzal, hogy mi *nem* szöveg, mint azzal, hogy mi az. Ha erről nem elhivatott nyelvészekkel, hanem az utca emberével váltunk szót, nem tapasztalunk különösebb nehézséget. Szöveg az, amit el lehet mondani, fel lehet olvasni, meg lehet hallgatni, le kell írni, amivel lehet helyettesíteni tetteket, vagy eltakarni gondolatokat. Kibernetikai terminológiában fogalmazva, a szöveg értelmezhető úgy, mint a *létezőnek a* kódolata, és a kód a *nyelv*.

A számítógépes szöveg alapvető sajátossága, hogy *ab ovo* performancia, azaz szubsztanciális eleme az, hogy létezik. Esetében nem az az érdekes, ami lesz vagy lehet, hanem az, ami van. A keletkezés ráadásul nem szubsztanciális része, mert számítógépes szöveg, feltevésünk szerint, úgy keletkezik, hogy egy „humán" szöveget kódolnak, azaz valamilyen módszerrel beviszik a számítógépbe. (NB – Minden bizonnyal vannak módszerek, melyekkel a számítógép maga is képes szöveget létrehozni.[R4.] Ennek lehetőségét nem kívánjuk elemezni, el tudjuk képzelni, de vannak bizonyos kételyeink a szuverenitást illetően.)

Szöveg és számítógép

Az alábbiakban azt próbáljuk meghatározni, hogy mi a szerepe a szövegnek a számítógépes rendszerben.

A kommunikáció során az adó két dolgot tesz: létrehoz egy olyan *formát*, amelyről feltételezi, hogy azt a vevő adekvát módon fogja értelmezni, és ebbe a formába tölti be a közvetíteni szánt dolgot, amit szokás *eszmének* [Beugrande-Dressler, 52. p.] is nevezni, de az informatika üzletágában azonban széleskörűen elterjedt egy egyszerű és kézenfekvő megnevezése: a *tartalom*.

A számítógép, mint a kommunikáció eszköze, ezt a tartalmat jeleníti meg, különböző formákban, melyeket itt performanciatípusoknak fogunk nevezni.

A továbbiakban a számítógépen közvetített szöveg alapformáit három rétegre osztva vizsgáljuk. Az első réteg a tartalom, a második a számítógép belső információábrázolása, a harmadikat pedig a performanciatípusok, amelyekben a tartalom manifesztálódik.

1. réteg: A tartalom

A számítógépen található (tárolt, feldolgozás alatt álló stb.) szöveg sajátos függőségi rendszerben jelenik meg. Ezt a jelenséget Karsai Róbert ekképp jellemzi: „...az írott és nyomtatott szövegek és elektronikus társaik között érdemes egy újabb elvonatkoztatási szintet feltételezni, és arra, hogy ez az elvonatkoztatási szint nemcsak a számítógépes szövegek megjelenése óta létezik..." [Karsai, 1. p]. Amit Karsai – az írott szöveg vonatkozásában teljes joggal – absztrakt szövegnek nevez, azonos a *tartalommal*.

Ez a nem manifesztált, potenciális szövegforma nem a számítógép sajátossága, hanem valahol az élőbeszéd, a hagyományos írásos szöveg hátterében is megtalálható.

Abban a pillanatban, amikor a beszédet kifejlesztő ember képes volt az időben és térben egyaránt átlátható szituáción túl is információt adni, azaz a közös észleletekhez kötött interakción túl megjelent a narráció képessége, szükségképpen megjelent a nyelvileg megalkotandó szöveg terve is. (Ezt nevezi van Dijk *eszmének* [[Dijk], in Beugrande-Dressler, 52]). Amit egyszer kimondtak, amit egyszer leírtak, az már létezik. Úgy mondhatnánk, hogy a tartalom a performálatlan, a performálandó szöveg.

2. réteg: A digitális szöveg

Neumann javaslata alapján a számítógép egységes, digitális kódot használ, mind a feldolgozandó tartalomhoz, mind a feldolgozó eszközökhöz. Függetlenül attól, hogy ez a tartalom miképp performálódik, a számítógépen minden szöveg alapvetően bitek sorozata.

(Tetszetős dolog lenne azt mondani, hogy ez így van ellenkező irányban is, azaz minden bitsorozat, minden bináris kódolat szövegnek tekinthető [R5.]. Ez azonban még átgondolandó, bizonyításra szorul.)

A szöveg digitális kódolása alapvetően a számítógépen való ábrázolásával függ össze, de az a tér, amelyet a bináris kód alkot, messze túlterjedt az egyes számítógépeken. Az internet felfogható a bináris kódolatok hálózatának is. Minthogy azonban az internet mégis összekapcsolt számítógépek szövedéke, a világhálón terjedő, elérhető információ *anyagát* tekintve homogén: minden digitális, minden bináris kódon valósul meg.

Ha lehetőségünk lenne a maga fizikai formájában megtekinteni a szöveg mögött levő bináris jelsorozatot, aligha tudnánk felfedezni, hogy annak mely része a nyelvi értelemben vett szöveg. A semleges kinézetű bitek egy szabályrendszer szerint alkotják a szöveget, amely meghatározza a folytonos jelfolyam *szegmentálását*, és egyezményes *jelentést* rendel az egyes szegmentumokhoz. Ez lényegében megegyezik a hagyományos írással: meghatározott bitcsoportok meghatározott helyzetben egy-egy betűk, írásjelet jelölnek, ahogy a hagyományos írásban ezt a szerepet grafikai kombinációk játszották el. A számítógépes szöveg többlete az, hogy a monotonan kattogó bitek bizonyos kombinációi nem szöveget kódolnak, hanem meghatározott akciókat, melyek a maguk helyén végbemennek, és létrehozzák a közölendő szöveg kívánt-tervezett formáját, illetve befolyásolhatják a megjelenés körülményeit.

3. réteg: Performanciatípusok

A számítógép mint szöveggenerátor lényegesen több kifjezőeszközt bocsát az író ember rendelkezésére, mint ami a korábbiakban ismeretes volt. A lehetőségek majdhogynem végtelenek, azonban az absztrakt szöveg még nem kommunikációképes. A gép–gép kommunikációban közlekedő szövegek emberihez tételére olyan eszközöket kell a számítógép rendszerébe foglalni, amelyek képesek betölteni a gép–ember interfész szerepét. Az ilyen eszközöket három csoportba sorolhatjuk:

a) szövegvizualizáló eszközök,
b) szövegfelolvasó eszközök,
c) szövegre reagáló eszközök.

A jelen cikkben ezek közül csak az első csoporttal foglalkozunk. A szövegfelolvasó eszközök [R6.] egyértelműen idetartoznak, mivel azonban e cikk szerzője nem rendelkezik kellő ismerettel e téren, – jelentőségük hangoztatása mellett

– most nem lesz szó róluk. A szövegre reagáló eszközök sorában a legjelentősebb maga a számítógép, amennyiben elfogadjuk, hogy a gépi kód maga is szöveg [R6.]. Ez izgalmas és fontos kérdés, de érdemében túllép a rendelkezésre álló kereteken.

A szövegmegjelenítés tekintetében a számítógép eszközkomplexumán belül meghatározó jelentősége van a perifériának. Hiába képes a bináris kód az emberi fantáziát messzemenően követni, a vágyak megvalósítósága a megjelenítőeszközökön múlik. Ismeretes, hogy a nyomdai szintű tipografikus szöveg előállítása a grafikus nyomtatók megjelenése óta lehetséges, és idősebbek arra is emlékzenek azokra az időkre, amikor például egyetlen grafikus display volt Magyarországon, a GD'71. A fejlődés eredményeképp a számítógép megjelenítőeszközei lényegében képesek a hagyományosnak tekintehtő vizualizáló technikák (kézi technikák, nyomda, film, televízió) szintjén szolgáltatni szöveget, illetve bizonyos tekintetben túl is lépnek azokon.

A megjelenített szöveg és a kód viszonya: a valódi szöveg és a képszöveg

A számítógép megjelenítőeszközei napjainkban lényegében kizárólag grafikusak. Aki ezt a szöveget olvassa, nem tudja megállapítani, hogy ez nem szövegként megjelenő kép-e (nevezzük ezeket a továbbiakban képszövegnek), hanem valódi szöveg (legyen ez a továbbiakban a szövegszövegnek).

Ez a különbség a szöveggel végezhető manipulációk terén jelenik meg: a szövegszöveg *szerkeszthető* (bővíthető, törölni lehet belőle, módosítható), emellett bizonyos mértékben a vizuális megjelenítése is módosítható (nagyítható, kicsinyíthető), ezzel szemben a képszöveg nem szerkeszthető, viszont olyan műveletek hajthatók rajta végre, mint a képeken (torzítások, elforgatás, lágyítás, szabdalás). Ez utóbbiak egy része a szövegszövegen nem végezhető el. A szövegszöveg szerkeszthetősége a kommunikáció nyelvi lehetőségei vonatkozásában fontos, kutatásra érdemes újdonságokat vet fel [R7.]

A képszöveg esetében a számítógépes szövegszerkesztők gazdag lehetőségei – itt a megjelenítésen túli felhasználásra gondolva – rendkívüli mértékben leszűkülnek. Az ismert nyelvtechnológiai eljárások, a szöveglétrehozás, illetve a szövegfeldolgozás számítógépes támogatása csak a szövegszövegen lehetségesek. Ráadásul a képszöveg általában nagyobb terjedelmű, így a szövegtovábbítás tekintetében is hátrányosabb.

A képszöveg előnye, hogy a bevitele egyszerűbb – lapolvasón könnyedén beolvasható, és a számítógépek stabilabban kezelik: míg a szövegszöveg jelentős mértékben hardverfüggő (pl. a szövegszerkesztők élesen reagálnak a telepített nyomtatók képességeire), a képek esetbén ilyen függőség lényegében nincs.

A nyomtatott szöveg

A számítógéppel kinyomtatott szöveg tulajdonképpen nem különbözik a hagyományos tipográfiai szövegtől. A nyelvhasználat szempontjából a különbséget a létrehozás jelenti. Ennek kapcsán mindenképpen fel kell figyelni a tipográfia megújult szerepére. Erre a jelenségre hívja fel a figyelmet Karsai Róbert: „Ma mindenki tipográfus... A számítógépen tárolt megjelenés-központú szöveg ... előállítása során nekünk kell gondoskodnunk a tipográfiai konvenciók betartásáról..."[[KARSAI] p. 19]. Ebből következően a tipográfia alkalmazása a számítógépes írás része lett, ami felveti annak kérdését, hogy a nyelvészet meddig és milyen mértékben tarthatja magát távol a tipográfiától.

Az, hogy az író embernek a számítógép rendelkezésére bocsátja a tipográfia eszközei, egyelőre nem jelent előrelépést. Úgy is mondhatnánk, hogy a nyelvi kompetencia nem párosul a tipográfiai kompetenciával. A számítógépes eszközök rendelkezésre bocsátanak egy sor tipográfiai eszközt, ez azonban csak lehetőség, a tényleges eredmény a felhasználón múlik. A gyártók elősegíthetik az ezzel kapcsolatos közösségi szintű megállapodásokat, azonban kikényszeríteni azokat nem képesek.

A vetített szöveg

Vetített szöveg az, ami a képernyőn vagy vetítővásznon jelenik meg az olvasó előtt. Egyik leglényegesebb sajátossága, hogy dinamikája révén megtöri a tipográfiai szöveg egy meghatározó korlátát, a linearitást. Ennek elsőrendű eszköze az úgynevezett hiperszöveg, az a lehetőség, hogy a megjelölt helyekről el lehet ugrani a szöveg, vagy egy másik szöveg megadott helyére, vagy akár egy eltérő megjelenítési közegbe (pl. internetes helyre).

Ezen túl a vetített szöveg lehetőségei kibővülnek: mozgó funkciók, az úgynevezett animációs megjelenítés, valamit a hangeffektusok a korábbiakban szövegben nem alkalmazott metakommunikációs eszközöket biztosítanak.

A vetített szöveg végül ki tud lépni tulajdonképpeni közegéből, a multimédiának nevezett kommunikációs formában egy lesz az egyenlők között.

4. réteg: A „kiterített" szöveg

A számítógépes szövegek ábrázolásának negyedik módja tulajdonképpen technológiai célt szolgál. A performált szöveg alapvetően két rétegű: az alap a hagyományos betűírás megjelenítése számítógépes eszközökkel, amelyet metakommunikációs effektusok egészítenek ki. A nyomtatott szöveg esetében ilyen effektusok a betűformák, a kiemelések (dőlt, félkövér, színezés stb.), illetve a szöveg tagolása. A vetített megjelenítésben ezeken túl különböző animációs hatások is alkalmazhatók.

A kiterített szöveg létrehozására külön nyelvosztály szolgál, a jelölőnyelvek osztálya. Lényegük az, hogy a megjeleníteni kívánt szöveg belsejében a számítógép által értelmezhető utasításokat helyeznek el (a korábbiakban szó volt az akciókat kiváltó számítógépes szövegekről!) . Ehhez pontosan ugyanazt az eszközt használják, mint a szöveg írásához, azaz ezeket a parancsokat is betűkkel-írásjelekkel ábrázolják. A jelölőnyelven írt szöveg közönséges betűsorozat, a normál szövegtől bizonyos tagolási szabályok alkalmazásával tér el.

Talán nem felesleges megjegyezni, hogy a ma ismeretes jelölőnyelvek kialakulását szokás a HTML megjelenéséhez kötni, de ez tévedés. Az ötlet az 1960-as években vetődött fel [CAMERON], a UNIX operációs rendszer író-dokumentumkezelő funkcióinak kifejlesztése kapcsán [DUNNE]. A kezdeti alkalmazások sorában minden bizonnyal a TEX nyelv a legismertebb, melyed Donadl E. Knuth definiált 1977 körül. [THOMPSON].

A jelölőnyelvek filozófiája azonban igencsak emlékeztet az úgynevezett rövid programokéra, amelyet a számítógép hőskorában arra alkalmaztak, hogy „egy (számító)gép utánozza egy másik gép viselkedését"[Neumann, pp. 333-336].

A jelölőnyelven létrehozott szöveg metaszövegnek tekinthető, amelyik egyik oldalról tartalmazza a szöveg írójának a szándékait, másik oldalról pedig áthidalja a számítógépek megvalósítási különbségeiből származó inkompatibilitást. A jelölőnyelv

átmeneti réteget képez a gépi kód és a performált szöveg, a gép kínálta lehetőségek és a szöveg írójának alkotó szándéka között.

Összefoglalás

Ennek az előadásnak két célja volt. Az egyik olyan közismert tények összefoglalása, amelyek a számítógép és a szöveg fogalmainak metszetében valósulnak meg. E tények együttes szemlélélése egy sor olyan kérdést vet fel, – ez a második cél – amelyek megválaszolása

a) a számítástechnika oldaláról tisztázhat egy sor olyan – inkább érzett, mint tudott – problémát, amelyek főképp a technika virtualizálódásával, az internet által kialakított sajátos globalizációval függenek össze;

b) a szövegtan területén pedig lehetővé teszi, hogy bizonyos túl bonyolult folyamatokat a számítógép modellként való felhasználásával képesek legyünk megközelíteni, megvalósítani a tudományos szemléletnek azt az integrációját, amelynek Neuman János fentebb idézett, A számológép és az agy című tanulmánya.

Irodalom

[Beugrande-Dressler] Beugrande, Robert de, Dressler, Wolfgang: Bevezetés a szövegnyelvészetbe. Corvina, É.n.

[Cameron] Cameron, Robert D.: Markup and Style: History and Philosophy. January 13, 2003 http://www.cs.sfu.ca/~cameron/Teaching/470/markup1.html, in. Computing Science, http://www.cs.sfu.ca

[Dunne] Dunne, Paul E.: A Brief Introduction to Text-Formatting using troff. http://www.csc.liv.ac.uk/~ped/teachadmin/troff_intro.html

[ÉKSZ2] Magyar értelmező kéziszótár, Akadémiai Kiadó, Budapest, 2003.

[Karsai Róbert] Absztrakt szöveg: az elektronikus szövegek alapjai. http://magyar-irodalom.elte.hu./robert/szovegek/absztrakt/

[MSZ7788/1] Az adatfeldolgozás fogalommeghatározásai és többnyelvű szótára. Szabványkiadó, Budapest, 1984.

[Neumann]Neuman János: A számítógép és az agy. In Neuman János Válogatott írások. Tipotex, Budapest, 2003.

[Petőfi, SZSZM] Petőfi S. János: Szöveg, szövegtan, műelemzés. Országos Pedagógiai Intézet, Budapest, 1990.

[Reboul-Moeschler] Reboul, Anne-Moeschler, Jacques: A társalgás cselei. Osiris 2000

[Thompson]Thompson, Skylar: LaTeX as an Alternative to Conventional Word Processing Programs. The History of TeX & LaTeX. http://www.earlham.edu/~thompsk/final_project/latex/node4.html

[Dijk], Teun van: Some aspects of text grammmar. The Hauge: Mouton. in [Beugrande-Dressler]

The Four-Level Model of Electronic Text

Ádám Kis
SZAK Publishers and the
Institute for Library Science and Computing of the
Faculty of Arts of ELTE University, Budapest
adam.kis@szak.hu

The purpose of this paper is to summarize well-known facts that are relevant to the relationship between the notions of text and the computer. When considering these facts simultaneously, several questions might be raised that, if answered from the aspect of the computer, may clarify several problems related to the increasing virtuality of technology and the particular globalization caused by the Internet. These problems are rather 'sensed' than 'known'. As for the field of textology, answering these questions enables us to utilize the computer as a model for some complex processes.

The computer, or rather the binary code representing the computer is related to text in many aspects. However, this question is too diverse and complex to investigate in the time frame for the presentation, so the paper is restricted to texts stored on or processed with the computer. These texts can be grouped into four levels, namely the following:

a) content or 'abstract text' that lies in the background of all other types of text and is becoming increasingly independent as computers integrate together and the network is becoming more and more general;
b) machine code: the substantial means of representing text on the computer, the 'raw material' of different manifestations of (textual) performance;
c) different types of (textual) performance: these are the various presentation of texts, rendered for human senses, i.e. printed, displayed on screen or on a projector;

These three levels of text form a strict hierarchy and together they form the technology of computational text representation that fits among the traditional types of text representation. The fourth level in question is not meant for the 'public', however, we can suppose that it facilitates the investigation of the nature of computational text to a great extent. This textual level is

d) the so-called 'projected' text: this means text written in mark-up languages.

This paper tries to look beyond this restricted subject. To achieve this, the author has marked points where the paper mentions a topic that requires further explanation or investigation.

Javaslat a magyar igék szemantikájának számítógépes implementációjára

Varasdi Károly[1] és Gábor Kata[1]

MTA Nyelvtudományi Intézet, Budapest, Benczúr u. 33.
{varasdi,gkata}@nytud.hu

Kulcsszavak szemantikai reprezentáció, igei argumentumszerkezet, konceptuális struktúrák, ontológia

1 Bevezetés

Mivel a szavak és kifejezések jelentése tetszés szerint, gyakorlatilag korlátlan mértékben finomítható, egy **teljes**, azaz hiánytalan szemantikai jellemzésekből álló szótár célkitűzése irreális lenne. Sőt, valószínűleg elméletileg sem lehet megadni egy olyan szemantikai leíró nyelvet, amely — amellett, hogy formalizálható —, elegendő kifejezőerővel rendelkezik a természetes nyelvek jelentésárnyalatainak tökéletes megragadására.

Ha a teljesség nem is tűzhető ki egy számítógépes alkalmazás szótári adatbázisát illetően, a **helyesség** követelménye igen. Egy kifejezés jelentésének leírása akkor helyes, ha a kifejezés *valóban* rendelkezik a leírás által neki tulajdonított jelentéselemekkel, legyenek azok bármilyen általánosak is. A szótár tekintetében ez azt jelenti, hogy egy helyes jelentésleírásokból álló szótárból soha nem kell visszavonni tételeket, és a szótár fejlesztése — az újabb elemek felvételén kívül — csak a már meglévő jelentésleírások további specifikálásán keresztül, azaz **monoton** módon történhet.

A fentiekből az következik, hogy kezdetben egy egyszerű, de rendkívül általános alapelemekből álló leírónyelvet érdemes definiálni, és a tényleges fejlesztés során ezt a nyelvet kell folyamatosan bővíteni a felmerülő igények szerint. A szemantikai leírónyelv legáltalánosabb alapelemei alkotják az **ontológiai alaptípusok** halmazát.

A szemantikai reprezentációs nyelvnek olyannak kell lennie, hogy a vele készített jelentésleírásokból gépi úton adekvát, azaz a természetes nyelvben is megtalálható következtetéseket lehessen levonni. Az ilyen következtetések levonása a jelentésreprezentációk között fennálló logikai kapcsolatokon alapul. Ezek a kapcsolatok **jelentésposztulátumok** útján valósulnak meg, amelyeket azonban *minél magasabb szinten kell kimondani.* Ez egy *hierarchikus* lexikonkoncepciót implikál, amelyben a specifikusabb elemek az általánosabb elemektől kapják meg **öröklődés** útján jelentésük egy részét (vagy akár egészét).

Az alábbiakban egy ezeknek az elveknek megfelelő, igen általános jelentésreprezentációs nyelvet mutatunk be, amely alapvetően Ray Jackendoff elképzeléseire

épül.[1] E keret természetesen nem helyettesíti a részletes lexikográfiai munkát, hiszen az igék jelentésének csupán a legfelső, legáltalánosabb szintjét ragadja meg. Azonban már ezen az igen általános szinten is releváns összefüggések válnak megfogalmazhatóvá.

2 Ontológiai alaptípusok

Jackendoff szerint a természetes nyelvek által használt legáltalánosabb szemantikai kategóriái a következők: THING (dolog), PLACE (hely), PATH (pálya), EVENT (esemény), ACTION (cselekvés), MANNER (mód), STATE (állapot), PROPERTY (tulajdonság), és AMOUNT (mennyiség).

Ez a bázis kategóriahalmaz szükség esetén azonban bővíthető. Például a későbbiekben szükségünk lesz a pálya kategóriáját finomabban differenciáló direkcionális alkategóriára (DIR), mivel egyes ragok jelentését kényelmesebben le lehet írni akkor, ha "irányított pályáról", azaz vektorról is tudunk beszélni.

Ezen kategóriák (vagy egy részük) elvileg kinyerhető a WordNet természetes nyelvi ontológiából (Fellbaum 1998), így azok manuális bekódolása elkerülhetővé válik.[2]

A fenti **ontológiai kategóriákon** kívül Jackendoff bevezet ezen kategóriák között működő **függvényeket** is. Mind az ontológiai kategóriák, mint pedig a közöttük értelmezett függvények rendkívül absztrakt entitások, amelyek a szemantikai mélyszerkezetben (Jackendoff terminológiája szerint a **konceptuális struktúrákban**) foglalnak helyet. A természetes nyelv szavai jelentés szempontjából *típusozva* vannak, továbbá szemantikailag vagy *egyszerűek*, s ez esetben jelöletük valamelyik ontológiai kategóriába esik, vagy *komplexek*, amikor is egy függvényekből és alapkategóriákból felépített összetett függvényt jelölnek. Például, a *house* szó szemantikai kategóriája THING, a *to the house* kifejezés egy PATH típusú entitást jelöl, míg magának a *to* prepozíciónak a jelentését — Jackendoff szerint — úgy lehet ábrázolni, hogy $[_{\text{PATH}}\ \text{TO}([_{\text{THING}}\ x])]$, ahol a félkövérrel szedett TO szimbólum már egy szemantikai *függvény* neve, és nem azonos a *to* prepozícióval (nevezhetnénk F_{213}-nak is).[3] Ennek alapján tehát az egész *to the house* kifejezés a $[_{\text{PATH}}\ \text{TO}([_{\text{THING}}\ \textit{house}])]$ szemantikai reprezentációra képződik le.[4]

Mivel a magyarban a prepozíciók szerepét a ragok és a névutók látják el, a megfelelő szemantikai szerkezetek is azokhoz kapcsolódhatnak. Ez a ragok esetében egy érdekes szakadást hoz létre a morfológia (szintaxis) és a szemantika

[1] Ld. (Jackendoff 1987) illetve (Jackendoff 1990). Jackendoffról van egy magyar nyelvű összefoglaló fejezet a (Kálmán–Trón–Varasdi 2002) kötetben is.

[2] Bár léteznek bizonyos előzmények, pl. (Dorr–Katsova 1998), ebben az irányban további kutatások szükségesek.

[3] A fenti zárójeles jelölést a továbbiakban is használni fogom: $[_{\text{OUTPUT}}\ \textbf{FUNC}([_{\text{INPUT}}\ x])]$ azt jelenti, hogy a FUNC függvény egy INPUT ontológiai kategóriájú x entitást vár, és ebből egy OUTPUT ontológiai kategóriájú entitást képez. Természetesen megengedünk egynél több argumentumú függvényeket is.

[4] Az (eltűnő) névelőről, s általában is: a determinánsok szerepéről Jackendoff nem ad számot. A kvantorok kezelésének hiánya rendszerének egyik komoly hiányossága.

között. Amíg ugyanis az angolban a prepozíciók teljes NP-khez járulnak, addig a magyarban a ragok morfológiailag egy N-hez járulnak; szemantikailag azonban mind az angol prepozíciók, mind a magyar ragok a teljes kvantoros főnévi csoport szemantikáját módosítják (pl. *a minden házhoz odament* mondatban az alany minden ház esetén bejárta az ahhoz vezető pályát, míg *a néhány házhoz odament* mondatban csak néhány ház esetében).

3 A lokalizmus filozófiája

Lokalizmusnak hívják a kognitív szemantikában azt a hipotézist, amely szerint a természetes nyelv kifejezőerejének magját a térbeli kapcsolatok kifejezései alkotják, s az absztraktabb (metaforikus) jelentések e magból érhetők el. A lokalizmus felfogása szerint az olyan térbeli vonatkozású kijelentések, mint például

(1) A hamutartó a földre esett

és az olyan átvitt értelmű kijelentések, mint a

(2) A dollár ára 200 Ft-ra esett

között komoly szerkezeti azonosság van. Míg ugyanis az első esetben egy fizikai tárgy változtatta meg térbeli pozícióját oly módon, hogy egy magasabb pozícióból egy alacsonyabba került, a (2) mondat esetében egy absztrakt entitás, a dollár ára végzett hasonló mozgást egy absztraktabb térben, a valutaárfolyamok terében. Sőt, az ilyen párhuzamosságokhoz nem is kell, hogy ugyanaz a kifejezés szolgáljon a változás megjelenítésére. A következő példában a varázsló az esemény elején emberi formát foglal el, a végén pedig megérkezik a sólymok formájához tartozó pontba, a személyes azonosság absztrakt terében.

(3) A varázsló sólyommá változott.

A lokalisták mindhárom esetben olyan pályákat feltételeznek ((1) esetében fizikait, (2) és (3) esetében absztraktat), amelynek a szóbanforgó entitások általi bejárása konstituálja magát a mondatban leírt eseményt.

Ennek a felfogásnak egy rendkívül előnyös vonása, hogy általános posztulátumok segítségével képes a három kijelentésben közös következtetési mintákat megragadni.

(4) A hamutartó a földre esett. *Tehát*: a hamutartó a földön van.

(5) A dollár ára 200 Ft-ra esett. *Tehát*: a dollár ára 200 Ft.

(6) A varázsló sólyommá változott. *Tehát*: a varázsló (most) sólyom.

A megfelelő posztulátum pedig az lehetne, hogy ha valami egy esemény keretében bejár egy (fizikai vagy absztrakt) pályát, akkor az esemény végén a pálya végpontjában található.

$$\textbf{PERF}([_{\text{EVENT}}\ \textbf{GO}([_{\text{THING}}\ x], [_{\text{PATH}}\ \textbf{TO}([_{\text{PLACE}}\ y])])]) \Longrightarrow [_{\text{STATE}}\ \textbf{BE-AT}([_{\text{THING}}\ x], [_{\text{PLACE}}\ y])].$$

4 Alkalmazások

4.1 Egyértelműsítés

(7) A dollár kétszáz forintra esett.

A fenti, (7) mondat kétértelmű. Tegyük fel ugyanis, hogy valaki egy dollár bankjegyet kétszáz és ötszáz forintos bankjegyekre dobál, és azt számolja, hogy dobásainak mekkora hányada esik kétszáz forintos bankjegyre. A (7) mondat, egyik jelentésében e "kísérlet" egyik részeredményét írja le. A másik, nyilvánvalóbb értelme a (7) mondatnak persze az, hogy a dollár *árfolyama* kétszáz forintra esett.

A bemutatott jelentésreprezentációs keret alapján a mondat két értelme könnyen elkülöníthető, ha figyelembe vesszük a következő két posztulátumot is:

> Ha egy p pálya kezdőpontja a T kategóriába esik, akkor a pálya végpontjának, valamint összes belső pontjának is T-be kell tartoznia.
>
> T típusú pályát csak T-vel kompatibilis kategóriájú objektum járhat be.

Ha tehát a *dollár* szó jelentését annak fizikai értelmében vesszük ("bankjegy"), akkor a *esik* szó jelentésében implicite meglévő pálya *fizikai* pálya kell, hogy legyen; ekkor kapjuk meg az első említett jelentést. Ha azonban a *dollárt* *absztrakt* értelemben vesszük ("árfolyam"), akkor (7) második értelmét kapjuk.

4.2 Robusztus jelentéselőállítás

E szemantikai reprezentációk használatával robusztusabb jelentéselőállítás válhat lehetővé. Tegyük fel, hogy a rendszer nem rendelkezik az igekötők jelentésreprezentációjával, bár felismeri az igekötős ige alapigéjét. Egy törékeny, hiánytalan elemzést megkívánó eljárás szükségképpen megakadna a következő mondat *odasétál* szavánál:

(8) János odasétált az ablakhoz.

Ha azonban az elemzés során egy "mohó" jelentésépítő algoritmust követünk, azaz az egyesíthető komponenseket azonnal egyesítjük is, akkor a következő lexikális reprezentációkat feltételezve lehetővé válik a mondatjelentés felépítése a teljes elemzés végrehajtása nélkül is, a jelentésekben kódolt információk kihasználásával.

$$\text{ablak} \rightsquigarrow [_{\text{THING}}\ \textit{ablak}] \tag{4.1}$$

$$\text{-hOz} \rightsquigarrow [_{\text{DIR}}\ \textbf{TO}([_{\text{PLACE}}\ \textbf{NEAR}([_{\text{THING}}\ x])])] \tag{4.2}$$

$$\text{sétál} \rightsquigarrow [_{\text{EVENT}}\ \textbf{GO}([_{\text{THING}}\ x], [_{\text{PATH}}\ y])] \tag{4.3}$$

E lexikon alapján az *ablakhoz* jelentését azonnal elő tudjuk állítani:

$$\text{az ablakhoz} \rightsquigarrow [_{\text{DIR}}\ \text{TO}([_{\text{PLACE}}\ \text{NEAR}([_{\text{THING}}\ \textit{ablak}])])]$$

Ebben szerepel egy direkcionális összetevő, ami azonban azonnal egyesíthető a *sétál* jelentésében megtalálható pálya összetevővel, hiszen annak egyik komponense éppen ezt kívánja meg:

$$\textit{path} = \langle \textbf{Source}, \textbf{Goal} \rangle =$$
$$= [_{\text{PATH}}\ [_{\text{DIR}}\ \text{FROM}([_{\text{PLACE}}\ \text{NEAR}([_{\text{THING}}\ x])])], \underline{[_{\text{DIR}}\ \text{TO}([_{\text{PLACE}}\ \text{NEAR}([_{\text{THING}}\ y])])]}].$$

Annak ellenére tehát, hogy a rendszer *nem* ismeri az *oda* igekötő jelentését, a fenti jelentésreprezentációk halmaza már egyesíthető, és kiadja a kívánt

$$\begin{aligned}&\text{János az ablakhoz sétál} \rightsquigarrow \\ &\quad [_{\text{EVENT}}\ \text{GO}([_{\text{THING}}\ \textit{János}], \\ &\qquad\qquad [_{\text{PATH}}\ [_{\text{DIR}}\ \text{FROM}([_{\text{PLACE}}\ \text{NEAR}([_{\text{THING}}\ x])])], \\ &\qquad\qquad\quad [_{\text{DIR}}\ \text{TO}([_{\text{PLACE}}\ \text{NEAR}([_{\text{THING}}\ \textit{ablak}])])])])].\end{aligned}$$

mondatjelentés-reprezentációt.

4.3 Implementáció

Az igei jelentés reprezentációját természetesen csak akkor hasznosíthatjuk bármiféle alkalmazáshoz, ha a jelentésreprezentációban szereplő *szemantikai argumentumokat* valamilyen algoritmus segítségével azonosítani tudjuk a mondatban ténylegesen előforduló elemekkel—legvalószínűbben az ige *szintaktikai vonzataival*. Mivel rendelkezésünkre áll az igék szintaktikai vonzatait leíró lexikai adatbázis, a munka első fázisában a meglévő vonzatkereteket célszerű jelentésreprezentációkhoz társítani. A jelenleg is bővítés alatt álló adatbázis most körülbelül 3,000 (gyakoriság szerint kiválasztott) ige 10,000 féle vonzatkeretét tartalmazza; elképzelésünk szerint a kódolási munka első szakaszát ezeknek a vonzatkereteknek a szemantikai kibővítése alkotná.

A lexikai adatbázis az igei vonatkeret leírását dependencia-szabályok formájában kódolja: a táblázat egy rekordja a címszó mellett azokat az elemeket tartalmazza, melyek együttes előfordulását az ige előfordulása dominálhatja a mondatban. A jelentésreprezentációban az igei jelentést **függvényként** ábrázoljuk, melynek argumentumai a szintaktikai vonzatoknak felelnek meg, a szemantikai reprezentációban pedig vagy a kilenc ontológiai alapkategória valamelyikéhez tartoznak, vagy maguk is ezeken működő függvényként ábrázolhatók. Természetesen az első feladat így az igei vonzatként előforduló kategóriák elemeinek besorolása, melyre az ige függvényként való ábrázolása épül. Ezután az igei jelentés leírását célszerű az igéknek azzal a (nem túl népes) csoportjával kezdeni, melyek argumentumai nem hivatkoznak további igei jelentésre. Azt tapasztaljuk, hogy ezeknek az igéknek szintaktikai vonzatai többnyire **esetragos névszók**.

Az esetragnak a szintaxisban kétféle szerepet tulajdonítunk: a lexikai adatbázisban vonzatként szereplő elemek esetragját az igei szubkategorizáció részének tekintjük, a vonzatkeretbe nem illeszkedő névszók esetragját pedig önálló elemként kezeljük, mely a névszót szabad határozói szerepre teszi alkalmassá. Ebből következik, hogy a szemantikai reprezentáció felépítésekor az esetraggal ellátott *vonzatszerepű* névszó jelentését az őt szubkategorizáló ige reprezentációjától tesszük függővé, míg a *szabad határozói szerepű* esetragos névszó jelentését kompozicionálisan, a névszó ontológiai típusából és az esetragnak megfelelő szemantikai függvényből akarjuk felépíteni. Így tehát amikor az igei vonzatkeretekhez szemantikai reprezentációt társítunk, két lehetőség közül választhatunk: vagy megadjuk az esetragos névszói vonzat lehetséges ontológiai típusát, valamint az esetragnak *az adott igei szubkategorizációban* megfelelő függvényt, vagy az esetragot az adott pozícióban szemantikailag üresnek tekintjük, és a szerkezet reprezentációját teljes egészében az igének tulajdonítjuk. A mondatelemzés folyamatára nézve az első eljárásnak az a következménye, hogy az esetraggal már összekapcsolt és a szintaxisban vonzatként azonosított elemet a szemantikai reprezentáció felépítéséhez újra morfémákra kell bontani. Mégis előnyösebb számunkra, ha az igei vonzatkeretben kapcsolóként funkcionáló esetragokat nem fosztjuk meg a saját jelentésüktől, mert ez teszi lehetővé, hogy az alkalmazott keret egyik legelőnyösebb tulajdonságát kihasználva, általánosításokat fogalmazzunk meg bizonyos igeosztályokra. Azt feltételezzük ugyanis, hogy az igék—szemantikai tulajdonságaik szerint—olyan csoportokra oszthatók, melyek szintaktikailag is hasonlóan viselkednek, például azonos szemantikai argumentumaikat jellemzően ugyanazzal az esetraggal jelenítik meg. A szemantikailag egyszerű igék szintaktikai vonzatainak és szemantikai argumentumainak megfeleltetésének, mely a kódolási munka első fázisát jelenti, egyik legfontosabb célja az alapvető igeosztályok feltérképezése.

Hivatkozások

Dorr, Bonnie J. and Maria Katsova. Lexical Selection for Cross-Language Applications: Combining LCS with WordNet. In *Proceedings of the Third Conference of the Association for MT in the America's*, Langhorne, PA, pp. 438–447, 1998.

Fellbaum, Christiane (szerk.). *WordNet – An Electronic Database*, A Bradford Book, MIT Press, Cambridge, MA, 1998.

Jackendoff, Ray. *Consciousness and the Computational Mind.* A Bradford Book, MIT Press, Cambridge, MA, 1987.

Jackendoff, Ray. *Semantic Structures.* MIT Press, Cambridge, MA, 1990.

Kálmán László–Trón Viktor–Varasdi Károly (szerk.). *Lexikalista elméletek a nyelvészetben*, TINTA Kiadó, Budapest, 2002.

Proposal for the Computational Implementation of the Semantics of Hungarian Verbs

Varasdi Károly[1] and Gábor Kata[1]

MTA Nyelvtudományi Intézet, Budapest, Benczúr u. 33.
{varasdi,gkata}@nytud.hu

Keywords semantic representation, argument structure, conceptual structures, ontology

The syntactic description of Hungarian verbal argument structure proves to be insufficient for a semantically adequate treatment of verbs, although up-to-date research in computational linguistics seems to confirm the inevitability of semantic representation.

In our paper we make a proposal for how to enrich an already existing valency dictionary with linguistically appropriate semantic representation. The framework we implement is Ray Jackendoff's Conceptual Semantics: the lexicon consisting of CS represenations is monotonically extensible, it is hierarchical (more specific elements inherit a part of their meaning from more general elements), and the representation can be related to the WordNet ontology. Such meaning representations make it possible to draw adequate inferences the way people do while using natural language. One of the most advantageous properties of this framework is that it allows of capturing relevant generalizations over large classes of verbs.

However, applying a framework designed for dealing with English phenomena to Hungarian language raises several specific problems. The structure of Hungarian language does not make it possible to treat morphology, syntax and semantics as independent modules using each other's output in a pipeline process. The rich morphology of Hungarian language completes several tasks that are assigned to syntax in English.

As opposed to the picture of an independent semantic module interpreting the output of syntax, we hypothesize a bidirectional, dynamic interaction between syntax/morphology and semantics. The semantic representation consists of nine basic ontological types but this set can be extended as required. The denotations of the "semantically simple" words of natural language belong to one of these basic types, while "semantically complex" words are represented as functions that, besides the basic categories, may contain other functions. Syntactic analysis and mapping to the semantic representation take place in a parallel manner. In case of structural homonymy, the structure of the semantic representations associated with the possible senses helps the system to choose the correct syntactic analysis of the sentence. This approach offers a relatively simple treatment for certain types of metonymy and ambiguity.

Fordítás magyarról magyarra – azaz a megértő kapcsolat az állampolgár és a kormányzás között

Vámos T., Soós I.

Számítástechnikai és Automatizálási Kutató Intézet
Magyar Tudományos Akadémia

vamos@sztaki.hu, soos@sztaki.hu

Az elektronikus kormányzás (E-governance) túllép az elektronikus kormányzat határain: az állampolgár és az állam kapcsolatrendszerébe több interaktív részvétellel bekapcsolódva új utat nyit a demokrácia rendszeres gyakorlásához. Így valósítjuk meg az információs társadalomban a mai athéni agórát. Ehhez az ember-gép-ember kapcsolatokban realizálható közvetlen párbeszéd szükséges: megérteni a különböző kulturális háttérrel és kifejezőkészséggel rendelkező lakosokat a saját anyanyelvükön, és természetes módon megfogalmazott mondanivalójukat a jogi eljárásokhoz igazítva. Az esethez tartozó szükséges szövegkörnyezet felismerésével és kiemelésével döntést támogató rendszert szolgáltatunk mind az állampolgár, mind az ügyével foglalkozó köztisztviselő részére.

Munkánk alkalmazza a magyar számítógépes nyelvészeti kutatás eredményeit, mint például a MORPHOLOGIC nyelvi elemzőjét vagy a Szegedi Kalmár Informatikai Laboratórium szövegkorpuszának tapasztalatait.

Az elemző a különböző szövegvizsgálati komponensek hierarchikus struktúrájára épül, melyeket a jól ismert és alaposan kidolgozott tárgyi esetek tanulmányozásából állítottunk elő. Támogatja azt a speciális nyelvi megközelítést, amelyet a Kaposvári Önkormányzat kísérleti fázisban lévő projektjében is fel kellett használnunk.

Felépítése a következő témakörök köré épül: szótár és szóelemzés, a mondanivalót közelítő, legfeljebb 3-4 szavas töredékek, mondatok és kisebb mondatfüzérek, melyek karakterisztikusan jellemzik a szöveget. Megfelelő szűrőket alkalmazunk a nem megfelelő helyesírásra, szinonimákra, s a témához nem kapcsolódó, s a szótárban sem szereplő szavak speciális kezelésére.

Az elemzés az általános esettárból történő folyamatos visszacsatolás alapján működik. Ez az esettár tartalmazza a szabványos forgatókönyveket, illetve a korábban elemzett és lefordított eseteket egyaránt.
Kifejlesztettünk egy illeszkedést vizsgáló algoritmust, mellyel a megértést valamilyen feltételek alapján, a gyakorlati használhatóságot szem előtt tartva mérni tudjuk. A megértett (lefordított) szöveget ez alapján egy gyakorlati alkalmazáson alapuló döntéstámogató rendszerhez tudjuk csatolni.

Az előadásban bemutatjuk az alkalmazott elemző struktúrát és egy részletesebb megvalósítási példát.

Translation from Hungarian to Hungarian, i.e. understanding connection between citizen and governance

Vámos T., Soós I.

Computer and Automation Institute
Hungarian Academy of Sciences

vamos@sztaki.hu, soos@sztaki.hu

E-governance means more than e-government, that is a new way of citizen-government contact, a new road for more democracy, more participation, an information society realization of the Athenian Agora in our present sense.
That requires a man-machine-man direct communication and dialogue: understanding the vernacular of the citizens having different cultural background and expressive power, translating their free text into the schemes of legal procedures, recognizing the essential content of the case and providing a decision support for the citizen and for the civil servant dealing with the case concerned.

The project applies all available results of the Hungarian e-linguistic community, i.e. the analyzer developed by MORPHOLOGIC and the experience and corpus of the Szeged Kalmár Informatics Laboratory.

The analyzer is built on a hierarchical structure of textual components strongly directed by the well known and elaborated subject frames. That framing supports the special approach of the system which is in experimental phase in a local government project for the city of Kaposvár.

The structure is built on the following issues: vocabulary, fragments of maximum 3-4 words, having some directive power towards the content, sentences and smaller string of sentences characterizing the definite content. Appropriate filters work for non-correct spelling, synonyms, non-relevant words especially those not contained in the special vocabulary.

The analysis works in a continuous feedback procedure from the standard case memory. That is a list of standard procedures, earlier analyzed and translated cases.
A matching algorithm is developed for this purpose, a good match measured by some criteria means the understanding in our pragmatic sense. The understood (translated) text is linked to a decision support system based on legal practice.

The structure of analysis and a text-processing example will be presented.

Nyelvi elemek érzelmi töltetének vizsgálata és felhasználása természetes nyelvi dialógusrendszerben

Tatai Gábor[1], Laufer László[2]

[1] Department of Computer Science, University College London
Gower Street, WC1E 6BT, London, UK
g.tatai@cs.ucl.ac.uk
[2] AITIA Informatikai Rt., 1117 Budapest, Infopark sétány 1.
llaufer@aitia.ai

Absztrakt
Cikkünkben bemutatjuk az általunk fejlesztett BotCom beszélgető-rendszerbe integrált GALA érzelmi elemző és generáló alrendszert. Ennek funkciója az, hogy egy felhasználóval történő természetes nyelvi beszélgetés folyamatában a beszélgetés érzelmi rétegeit is figyelembe vegye a válaszadás során. A GALA többrétegű architektúrával rendelkezik, és az R. Plutchik féle érzelmi modellt használja. A rendszer jól használható rövid chat-dialógusok érzelmi rétegének kezelésére, melyből kinyert információval animált beszélgető partnert tudunk vezérelni.

Kulcsszavak: Affective computing, érzelmi modellezés, beszélgetőrendszer

1 Bevezetés

Kutatásaink során – mely részben természetes nyelvi dialógus-keretrendszer fejlesztésére fókuszál – szembesültünk a szövegek érzelmi töltetének meghatározásának és felhasználásának fontosságával. Ismeretes, hogy a számítógépes nyelvi alkalmazások elsősorban a nyelv, a mondatok szerkezeti, strukturális problémáira koncentrálnak. A nyelv formális leírásában, praktikus nyelvi alkalmazások készítésében számos magyar kutatónak és fejlesztőnek jelentős tapasztalata van. Az ember-számítógép interfészekben felhasználható számítógépes dialógusrendszerek esetében azonban felmerül az igény, hogy a szöveg által hordozott szintaktikai információ és szemantikai tartalom mellett az emberi kommunikációban jelentős szerepet játszó érzelmi réteg is detektálható, leírható, felhasználható és generálható legyen.

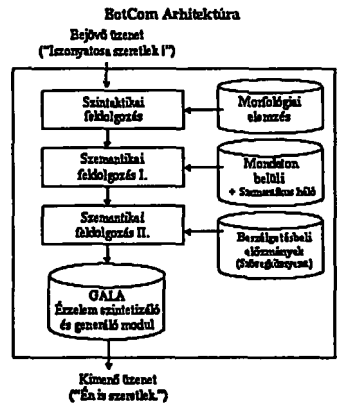
Az alkalmazás fontosságát több oldalról is megvilágíthatjuk. Az emberek sokkal könnyebben ismerkedhetnek meg a számítógép nyújtotta lehetőségekkel, ha kevesebbet kell tanulniuk azok használatát. A természetes nyelvi dialógusrendszerek olyan virtuális „beszélgetőpartnerként" használhatók, melyek a bevitt természetes nyelvi szöveges információt fel tudják dolgozni, és arra akár egy általános témájú beszélgetés keretében értelmes mondatokkal tudnak válaszolni, vagy a mondatokban szereplő esetleges utasításokat fel tudják ismerni, és végre tudják hajtani. A dialógusrendszerben így nem csak a gép vezérlése válik érthetőbbé, hanem a visszajelzés is jobban

értelmezhető a felhasználó számára – természetesen abban az esetben, ha a gépi válasz adekvát. (A diskurzusmenedzsment, a szemantikus információ kinyerése és követése stb. ismert és nehéz problémakör, melyet érintőlegesen tárgyalunk csupán. Egy másik hasznos alkalmazási területre példa az e-learning keretrendszerekhez illeszthető, sok esetben animált arccal kiegészített, természetes nyelvi kérdező-válaszoló rendszer, mely mind saját, mind külföldi kísérletek alapján megnöveli a tanulás hatékonyságát.

A természetesség érzetét növeli az, hogy ha a virtuális beszélgetőtárs a kommunikáció érzelmi szintjét is érzékelni tudja és ezt az információt válaszaiban kihasználja, például úgy, hogy hasonló vagy ellentétes érzelmi töltetű válaszokat próbál adni, a beállításoktól és helyzettől függően. Beszédfelismeréssel foglalkozó kollégáink számos információhoz jutnak az intonáció/prozódia vizsgálatával. Annak érdekében azonban, hogy egy szöveges információ érzelmi töltetét elemezzük, mellyel nem „jön" ilyen másodlagos információ, szükségünk van egy olyan adatbázisra, mely tartalmaz szavakhoz, kifejezésekhez rendelt érzelmi töltetet. Továbbá értelemszerűen egy kontextuselemző is elengedhetetlen annak érdekében, hogy érzékelni lehessen a szövegkörnyezet következtében fellépő érzelmi jelleg változását.

Cikkünkben bemutatjuk az erre a problémára született megoldásainkat, melyek természetesen még fejlesztés alatt vannak. Eddig eredményeink már számos nemzetközi publikációban megjelentek [2; 3; 4; 5], ezért fontosnak tartjuk azt, hogy a magyar szakmai közönség is jobban megismerje ezeket. A problémát egy szintetikus érzelemmodellező és kiértékelő rendszer, a GALA eszköz kifejlesztésével próbáljuk megoldani. Ez az eszköz tartalmaz egy Robert Plutchik által ismertetett geometriai alapú érzelmi modellt [6; 7], erre építettünk egy további réteget, amely Michael A. Gilbert üzenet-aktus *(message act)* elméletére [8] támaszkodva teremti meg a kapcsolatot a nyelvi elemek és az érzelemkifejezések között. A üzenet-aktus elmélet lényege, hogy a kommunikáció nyelvi elemei érzelmi információt közvetítenek, amely az ismert beszédaktus *(speech act)* [1] elmélethez hasonlóan több különböző komponenssel írható le (lokúciós, információs, illokúciós és perlokúciós aktusok). A két modell kiváló alap szintetikus érzelmek előállításához. A szintetikusságot azért tartjuk fontosnak hangsúlyozni, mert számos érzelmi modell létezik, továbbá emberenként is jelentős különbségek vannak, különösen a bonyolultabb érzelmek megjelenésében és működésében, ezért úgy gondoltuk, hogy egy egységes modell, mely ugyan számos egyszerűsítésen esett át, mindenképpen szükséges az érdemi felhasználáshoz. Ez azonban egyszerre jelenti azt is, hogy pontosan tisztában vagyunk azzal, hogy a GALA által felismert illetve generált érzelem nem feltétlenül feleltethető meg egy adott ember érzelmi állapotának, azonban tapasztalataink szerint statisztikailag egy jó általános, mesterséges érzelmi állapotváltozás modellezésére alkalmas. Elvégre valamennyien különbözőek vagyunk...

2 A BotCom beszélgetőrendszer felépítése



Beszélgetőrendszerünk többrétegű felépítésű, a kezdeti szakaszban az üzenetet magát, illetve annak szemantikai tartalmát azonosítjuk és csak a legutolsó szakaszban kerül sor az érzelmi töltet meghatározására. Az első szinten szintaktikai feldolgozás történik: az egyes szavak azonosítjuk be szótövesítés [5] és egyéb morfológiai elemző eszközök alkalmazásával. Eztán kerül sor az üzenetnek a párbeszéd-szekvenciákból álló tudásbázisunkkal való összevetésére. Itt a bejövő üzenet részeinek mind a mondaton belüli grammatikai szerepét, mind az egyes szavak jelentés tartományába eső egyéb kifejezéseket figyelembe vesszük. Ez utóbbit az OSZK Tezauruszával [9], illetve egyéb szabadon felhasználható szinonima szótárak segítségével végezzük. Az üzenet szemantikai tartalmának pontosabb eldöntése a következő szinten valósul meg. Az azonosítást előre tárolt témakörök kulcsszavaival végezzük el. Annak érdekében, hogy pontosabban meg tudjuk határozni az adott dialógus szakasz fő témáját, figyelembe vesszük a dialógus korábbi szakaszában szereplő közlések már beazonosított szemantikai tartalmát is. Ez igen lényeges momentum, hiszen a dialógus során fenn kell tartani egyfajta perzisztenciát a témát illetően, annak érdekében, hogy a beszélgető robot se ugráljon túlságosan gyorsan más témára, akár félreértés, akár proaktív működés következtében. Így nagyobb biztonsággal tudunk a felhasználó üzenetéhez jól illeszkedő válaszokat adni. Az érzelmi feldolgozás csak ezután következik a később ismertetett GALA modul segítségével.

2.1 A BotCom rendszer témakezelése

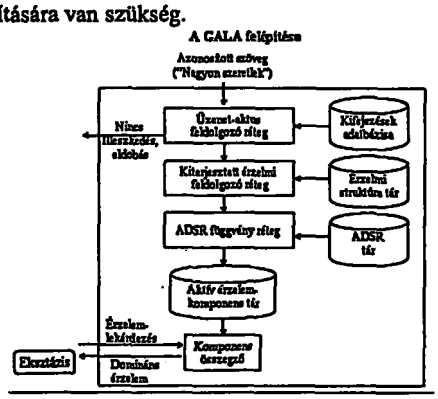
A BotCom válaszgenerálása igen nagymértékben függ a bejövő üzenetek helyes szemantikai beazonosításától. Ennek érdekében arra törekedtünk, hogy tudásbázisunk minél jobban lefedje a szokványos chat-en történő beszélgetés közben felmerülő témákat. A tudásbázist részben automatikusan, a weben szereplő chatlog fájlokból, részben 70 diák nyári-munka keretében történő, előre megadott témákról való beszélgetéseinek naplójából töltöttük fel. A feltöltésnél mind a témafelvetésekben, mind a válaszalternatívák varianciájában próbáltunk megfelelő számú változatot felhalmozni. Előzetes kísérletek az egy kérdésre adott 4-5 sablon választ elégségesnek mutatták ahhoz, hogy egy fél órás beszélgetés kellően változatos legyen. A helyes szemantikai tartalom kinyerésének másik sarokköve a bejövő üzenet előzményeinek, a dialógus *history*-jának nyomon követése. A korábbi feldolgozási szakaszok nyújtotta alternatívák közötti választásnál fontos szempont a beszélgetés menetébe, mind a kommunikációs funkció, mind a szemantikai szempontból való

kációs funkció, mind a szemantikai szempontból való illeszkedés. A rendszer ily módon tulajdonképpen elvárásokat alakít ki a felhasználó üzenetét illetően és ezen elvárások mentén keresi a megfelelő választ.
Beszélgető rendszerünk így kevésbé „téved" el a fókuszukat nem tartalmazó közlések következtében. Például:
BotCom: Hogy vagy?
Felhasználó: Boldog vagyok.
B: Ezt jó hallani.
F: Nagyszerű érzés elmerengni a lét forgatagának végtelen lehetőségein.
B: Örülök, hogy jól érzed magad.
Ugyanez a dialógus a history figyelmen kívül hagyásával így zajlana le:
BotCom: Hogy vagy?
Felhasználó: Boldog vagyok.
B: Ezt jó hallani.
F: Nagyszerű érzés elmerengni a lét forgatagának végtelen lehetőségein.
B: Milyen befektetési lehetőségeket ismersz?
A második példán látszik, hogy merengés, lé és a forgatás szavak együttállásának megfelelő szemantikai hálóval történő kiértékelése igen könnyen más síkra terelheti a beszélgetést. Míg ha a korábbi szekvenciák jelentésének tartalmát is figyelembe vesszük, a „nagyszerű érzés" sokkal jobban illeszkedik a korábbi szemantikus tartalomra, így érdemes megkockáztatni a mondat kevésbé illeszkedő részeinek akár a figyelmen kívül hagyását, és a „pozitív kedélyállapot" témában való válaszkeresés preferálását.

3 A GALA modul felépítése

A GALA érzelem szintetizáló és generáló rendszer a BotCom keretrendszerhez hasonlóan egy többlépcsős folyamat eredményeként nyeri ki a felhasználó üzeneteként azonosított mondatból a kommunikációs szándékot, illetve az érzelmi tartalmat. A modul önállóan is kezelhető és más keretrendszerekbe is könnyen integrálható, ahol írott szöveg érzelmi töltetének az azonosítására van szükség.

Az első rétegben az érzelmekkel és kommunikációs szereppel megfelelően felcímkézett adatbázisból kikeressük az azonosított, így az adatbázisunkban már meglévő szöveghez tartozó üzenet-aktust. Ha ez nem szerepel az adatbázisban, akkor értelemszerűen a modul nem gyakorol hatást a válasz kiválasztására. Ha megtaláljuk az adatbázisban az üzenetet, vagy néhány szegmensét, akkor ezeket továbbítjuk az érzelem feldolgozó rétegnek. Ez vagy az egész szöveget, vagy az egyes szegmenseit megkeresi az Érzelemi



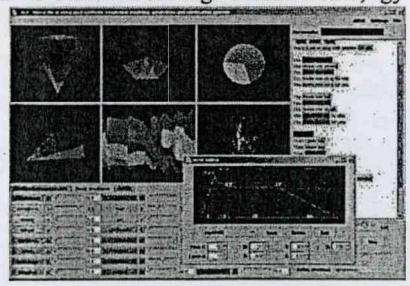
struktúra tárban, és kiértékeli a szöveg érzelmi töltetét. Ez a töltet, ahogy azt mindennapi beszélgetésekben is tapasztalhatjuk, nem feltétlenül csupán egyetlen alapérzelem, hanem több különböző erősségű alapérzelem együttes jelenlétével jellemezhető. Az érzelmek reprezentációját a Plutchik féle 24 alapérzelemmel operáló modell segítségével végeztük. A modellben több változtatást hajtottunk végre, illetve kiegészítettük egy minden érzelmet időbeli karakterisztikával jellemző függvénykezeléssel.

Az egyes érzelmekhez, illetve az egyes kifejezésekben hordozott összetett érzelmekhez (értsd több alapérzelem együttes jelenléte), különböző lefutású karakterisztikus függvényeket rendelünk. A függvények a könnyebb kezelhetőség kedvéért egyszerű ADSR (Attack Decay Sustain Release – Felfutás Csökkenés Kitartás Elengedés) egyenes szakaszokból állnak. Ezáltal konfigurálható, hirtelen hargú, de könnyen felejtő, vagy haragtartó, de nehezen feldühödő viselkedésű robot. A zenei hangok lecsengésének modellezésére is használt ADSR függvényekkel elérjük azt, hogy az egyes szöveg részletek által okozott hangulati hatások időben változnak, nem pillanatnyi a befolyásuk, és a korábbi üzenetek érzelmi töltete is hatást gyakorol a későbbi válaszok érzelmi töltetére. A komponens összegzés fázisához érve meghatározhatjuk egy adott pillanatban a korábbi, már lecsengőben lévő érzelmi hatások, és a legutolsó üzenet kiváltotta érzelmi hatások eredőjét. Ezt az eredő érzelmet leképezünk a már említett Plutchik-féle 24 alapérzelemmel operáló modellre, kinyerve a beszélgetés eddigi szakasza keltette domináns érzelmet. A robot a továbbiakban eszerint próbál viselkedni.

Mivel a BotCom rendszer már azonosította a bejövő üzenetet, és így kikereste az ehhez tartozó lehetséges válaszokat is, rendszerünknek csupán már az a dolga, hogy a válasz alternatívák közül a GALA által kiértékelteknek megfelelő kommunikációs tartalommal, illetve érzelmi töltettel rendelkezőt válassza ki az elérhetők közül és közölje a felhasználóval. Egyezés hiánya esetén neutrális, vagy a legközelebbi érzelmi töltetű választ adja a rendszer.

4 A GALA vizuális modellező felülete

A GALA modellező eszköz ennek megfelelően lehetőséget biztosít arra, hogy mondatokat, kifejezéseket, szavakat egy grafikus felületen keresztül érzelmi töltettel lássunk el. Továbbá tartalmazza az érzelmek dinamikájának kezeléséhez szükséges többrétegű konfigurációs lehetőséget. A GALA használatával felcímkézett adatbázist a dialógusrendszer fel tudja használni, a ki- és bemeneti szövegek elemzéséhez, így azok érzelmi tartalmának megbecsléséhez.



A modellező eszköz képes grafikusan ábrázolni egy-egy beolvasott szöveg (pl. vers, párbeszéd) érzelmi töltetének időbeli alakulását, ezáltal segítséget nyújt a címkézés vagy párbeszéd során előálló hosszabb szövegek, dialógusok elemzéséhez és egyben a szintetikus érzelmi állapot változásának

realisztikussági fokának meghatározásához. Az első lépés kiértékelését 65 magyar nyelvű vers/dal és 70 félórás chat-dialógusokból rögzített beszélgetés érzelmi címkézésével végeztük. Természetesen a címkézést végző személy értelmezésétől sosem lehet függetleníteni a rendszert, de előre kiadott irányelvek és egy személy által történő utólagos ellenőrzés elfogadható általános érzelmi jelölést eredményez.

A címkézés során bővülő érzelmi adatbázis egyre nagyobb mértékben teszi lehetővé ismeretlen szövegek automatikus felcímkézését, ugyanakkor ez még sok problémát hordoz magában. Hiába az alkalmazott több tízezer bejegyzést tartalmazó szótár, szinonimaszótár és szemantikus háló, az érzelmek címkézése elkerülhetetlen és időigényes, hiszen az egyes szinonimák számos esetben az általuk hordozott érzelmi töltetekben különböznek egymástól. E mellett sokszor rövid kifejezések címkézése szükséges ahhoz, hogy a valóságos érzelmi töltetet felfedjük, mely esetleg különbözik a kifejezés tagjai által önállóan hordozott érzelmi töltettől.

5 Összegzés

Ahogy ez várható volt, az egyszerűbb, egyértelműbb szavak, illetve világos, rövid mondatok esetén még viszonylag pontos találatokra képes a rendszer, ugyanakkor minél bonyolultabb az ismeretlen szöveg, annál gyakrabban ad kissé szürreális érzelmi reakciókat. Jelenlegi kutatásaink ezért arra irányulnak, hogy a GALA kontextuselemzőjének továbbfejlesztésével ezekben a komplikáltabb esetekben is jobb eredményt tudjon elérni.

Irodalomjegyzék

1. Searle, J. R. What is a Speach Act. In The Philosophy of Language, J. R. Searle, Ed. Oxford University Press, London (1979)
2. Tatai G., Csordás A., Kiss Á., Laufer L., Szaló A.: Happy chatbot, happy user. Proceedings of the 4th International Working Conference on Intelligent Virtual Agents (IVA'03), Irsee, Germany, Springer Verlag (2003).
3. Szaló A., Csordás A., Laufer L., Tatai G.: The GALA layered emotion model for advanced HCI interfaces. Proceedings of the 3rd International Conference on Hybrid Intelligent Systems (HIS'03), Melbourne, Australia, IOS Press (2003).
4. Tatai G., Csordás A., Szaló A., Laufer L.: The chatbot feeling - Towards a usable emotional model for Internet ECAs. Proceedings of EPIA'03 - 11th Portuguese Conference on AI, Beja, Portugal, Springer Verlag (2003).
5. Magyar Ispell/Myspell Szótármodul: http://www.szofi.hu/gnu/magyarispell/
6. Plutchik R.: The nature of emotions. American Scientist 89(4):344-350 (2001).
7. Plutchik R.: A general psychoevolutionary theory of emotion. Plutchik, R., Kellerman, H. (szerk.): Emotion theory, research, and experience, Theories of emotion, pp 3-33, Academic Press (1980).
8. Gilbert M. A.: Language, words and expressive speech acts. van Eemeren, F., Grootendorst, R., Blair, J. A., Willard, C. A. (szerk.): Proceedings of the Fourth International Conference of the International Society for the Study of Argumentation, pp 231-234 (1999).
9. Országos Széchényi Könyvtár Tezaurusz: http://www.oszk.hu/ujdonsag/tezauruj.html

Extraction of Affective Components from Chat Conversations and Their Use in Natural Language Dialogue Systems

Gábor Tatai[1], László Laufer[2]

[1] Department of Computer Science, University College London
Gower Street, WC1E 6BT, London, UK
g.tatai@cs.ucl.ac.uk
[2] AITIA Informatikai Rt., 1117 Budapest, Infopark sétány 1.
llaufer@aitia.ai

Keywords: Affective Computing, Emotional Modeling, Dialogue System

We are carrying out a research in the field of Human Computer Interaction and developing a natural language dialogue system in Hungarian. During the design and implementation of our chatterbot system we faced the importance of detecting the emotional load of incoming user messages and reacting on them appropriately.

In the beginning chapters we briefly describe the architecture of our dialogue system, BotCom with examples of its semantic processing capabilities. We give examples of how the system is handling the topics of the discussion, how the dialogue history is being used in order to enhance the reply generation.

In the subsequent parts we give an overview of the emotional state detecting, processing and generating module, called GALA, which is founded on the grounds of Robert Plutchik's emotional model. We describe the different layers of GALA, how Plutchik's 24 basic emotions are being used and how we extended the model in order to be able to create permanent, though dynamically changing emotional states. We show how BotCom is utilizing the detected emotional loads of the user's messages, therefore enabling the chatterbot to give relevant answers both semantically and affectively.

In the final chapter we explain how the database of GALA was filled up with expressions assigned to their emotional loads. We also describe a graphical user interface (GUI) being designed to model the changing emotional loads in dialogues, songs and poems. The GUI is being used for the emotional labeling of the phrases needed for the expression database of GALA.

Tudásalapú természetesnyelv-feldolgozás

Kálmán László[1], Balázs László[2] és Erdélyi Szabó Miklós[3]

[1] Alkalmazott Logikai Laboratórium, Budapest
MTA Nyelvtudományi Intézet, Budapest
MTA/ELTE Elméleti nyelvészet szakcsoport, Budapest
kalman@nytud.hu
[2] Alkalmazott Logikai Laboratórium, Budapest
bazsi@all.hu
[3] Alkalmazott Logikai Laboratórium, Budapest
MTA Rényi Alfréd Matematikai Kutatóintézet, Budapest
mszabo@renyi.hu

Absztrakt: A 80-as évek óa két fő irányzat van a számítógépes nyelvészetben: a statisztikus megközelítést használó „felszíni technológiák", valamint azok a próbálkozások, hogy a generatív nyelvelméletet mégiscsak használni lehessen. Az előző volt a sikeresebb, de a mondattan és főleg a jelentéstan területén nem volt előrehaladás. Az általunk javasolt alternatíva olyan hibrid rendszer, amelyben különböző tudásfajtákat különböző eszközökkel kezelünk. Az „automatikus" folyamatokat, mint amilyen a beszédfelismerés, statisztikus eszközökkel modelláljuk, míg a „tudatos" működéseket pedig, mint amilyenek a megértés és fogalmazás mélyebb szintjei, logikai eszközökkel, az ún. konstrukciós nyelvtan elméleti alapján, az abdukciónak nevezett okoskodási módszerrel.

1. Bevezetés

A múlt század 60-as éveinek nagy generativista fellángolását a 80-as évekre a csalódás hangulata váltotta fel. Reménytelenül bonyolultnak és inadekvátnak bizonyult a természetes nyelv feldolgozásának szabályalapú megközelítése. A 90-es évek óta nyilvánvaló, hogy a generatív nyelvtan merev és moduláris nyelvtanfelfogására nem lehet működő rendszereket alapozni, és elterjedtek a nagy korpuszok elemzésén alapuló statisztikus módszerek, amelyek az alsóbb nyelvi szintek (beszédlétrehozás és -felismerés, alaktani elemzés) tekintetében igen sikeresek is. A kutatók igyekeznek a korpuszalapú módszereket a magasabb szintekre, a mondattani és jelentéstani elemzés szintjére is kiterjeszteni.

A mi vállalkozásunk azon alapul, hogy — bár nem tagadjuk az automatikus tanulás és a nyelvfeldolgozás sztochasztikus, nem-determinisztikus modelljeinek fontosságát — a magasabb, többé-kevésbé tudatosan végzett nyelvi tevékenységekben a tudással és okoskodással való összefüggéseknek ugyanúgy fontosságot tulajdonítunk, mint a formai minták felismerésének. Ez nem a szabályalapú megközelítések „visszacsempészése" a nyelvfeldolgozásba, hanem annak az érvényesítése, hogy nagyjából tudatosan végzett, a gondolkodással és következtetéssel összefüggő emberi tevékenységeket szerintünk nem lehet ugyanolyan eszközökkel

modellálni, mint az olyan szinte automatikus, szinte a reflexszerű beidegződésig rutinszerű tevékenységeket, mint a hangképzést és a szóalakok felismerését, a szótárból való kikeresését.

A következőkben először visszatekintünk a természetes nyelv feldolgozásának fő fejlődési vonalára, és elhelyezzük az általunk tervezett rendszert ebben a vonalban (2.). Utána részletesebben kifejtjük a nyelvi megértéssel és a megnyilatkozások létrehozásával kapcsolatos „filozófiánkat" (3.), majd azt, hogy magát a nyelvi tudást hogyan modelláljuk (4.). Az utolsó pontban (5.) a rendszer továbbfejlesztésének, különösen tanulhatóvá tételének problémáiról szólunk.

2. Előzmények

Nagy volt a lelkesedés, amikor a 70-es években Chomsky és Montague előálltak a nyelvtan és a jelentéstan formalizálásának nagyszabású terveivel. Végre valaki precízen megmondja, mi is van a természetes nyelvekben — gondolták a számítógépesek, és már fenték a fogukat, hogy majd beszélgető- meg fordítógépeket fognak építeni.

A 80-as években már a csalódás volt az uralkodó hangulati elem közöttük. A javasolt formális modellek nemcsakhogy gyakorlatilag kezelhetetlennek bizonyultak — ezen még csak-csak lehetett volna javítani, változtatni —, de egyre inadekvátabbnak is tűntek. Bárhogy fájt is a generativista szívünknek, a fő baj ezekkel a modellekkel a **modularitásuk** volt. Ha a mondattan autonóm, és nem használ jelentéstani-pragmatikai információt, akkor az elemzés problémája szinte megoldhatatlan; ha a jelentéstan autonóm, és nem keveredik a pragmatikával, akkor reménytelenül használhatatlan és bonyolult. A moduloknak annyiféleképpen kellene együttműködniük, hogy már nem is nagyon nevezhetnénk őket moduloknak.

Kétféle ellenreakciót váltott ki az elkeseredett hangulat: az egyik a nyelvelméletre alapozott modellek teljes elvetését (ez a máig is sikeres statisztikus eljárásokhoz vezetett; jelszavuk: „szakítsunk a csak szakértők által feltölthető rendszerekkel, mérjük inkább ki, hogy mik a jó paraméterek"), a másik a modularitás és/vagy a futószalagszerű, szekvenciális modellek elvetését. Az utóbbi sajnos csak jelszavak formájában fogalmazódott meg („készítsünk integrált rendszereket", illetve „szakítsunk a tisztán procedurális felfogással", vö. [2]). A mi kutatásunk az utóbbi vonalba illeszkedik — bár nem tagadjuk, hogy bizonyos területeken a statisztikai módszerek a legjobbak.

Az általunk épített rendszer — amelynek még arra se volt ideje, hogy nevet kapjon — moduláris, de a modulok nem a nyelvészetből ismert rétegződésnek (fonológia, morfológia, szintaxis stb.) felelnek meg, hanem aszerint állítottuk fel őket, hogy milyen típusú eljárások működnek bennük. Tudásalapú rendszerről lévén szó, a „tudás" fogalmának, különböző értelmezéseinek felelnek meg ezek a modulok. A szó sokértelműségét könnyű belátni, elég, ha arra gondolunk, hogy az immunrendszerünkben is van „tudás", hiszen különböző fehérjetípusokat képes felismerni, elkülöníteni és reagálni rájuk, van ún. operacionális tudásunk (mint pl. a járás), amelyet nem tudunk expliciten megfogalmazni, mégis tudatosan

használjuk, van olyan tudásunk, amelyeket szabályok formájában is meg tudunk fogalmazni, és így tovább.

Amikor a számítógépesek „tudásalapú" rendszerről beszélnek, akkor a sokféle tudásféle közül arra gondolnak, amely a logikából ismerős „állítások + következtetési szabályok" sémával írható le, vagyis arra a tudásra, amely logikai állítások egy halmazából áll, plusz mindazokból a potenciális állításokból, amelyeket ebből a halmazból következtetésként le lehet vonni. Mi azonban, amikor a „tudásalapú" NLP szükségességét elismerjük, nemcsak erre a fajta tudásra gondolunk.

A mi tudásalapú rendszerünknek olyan moduljai is vannak, amelyek tudattalan, teljesen automatikus mechanizmusokat szimulálnak, ezeket technikailag nem is nevezhetjük „tudásalapúaknak", hiszen emberi megfelelőiket okoskodás nélkül, teljesen automatikusan használjuk. Ilyen mechanizmusok például a **célok** kitűzése, illetve a még absztraktabb, a célok mögött meghúzódó mechanizmusok, amelyeket **vágyaknak** nevezhetünk. Ezek irányítják, mozgatják végső soron a beszéd és a megértés folyamatát. De ilyen tudattalan mechanizmusok például a kiejtés és a szófelismerés folyamatainak nagy része is, ezek is olyan modulok, amelyekbe a tudatos gondolkodás, érvelés nem tud igazából „beavatkozni".

A másik véglet a tudatos okoskodás szintje; a tudatos modulok a mesterséges intelligenciából jól ismert, csak apróbb újításokat tartalmazó módszerekkel elsősorban **tervezést** végeznek. A konkrét helyzet, a cél és általában a külvilág ismeretében terveket dolgozunk ki a célunk eléréséhez. Fontos, hogy **részleges terveket** is használunk (amelyeknek nincs minden részletük kidolgozva, további, de elhalasztott tervezési folyamatot tesznek még szükségessé), valamint hogy **előregyártott tervek** (ún. **receptek**) is a rendelkezésünkre állnak, az olyan esetekre, amelyeket már egyszer megoldottunk, és emlékszünk a megoldásra. A tudatosság pillanatnyiságának megfelelően még ezeknek a tervezési folyamatoknak a nagy része is „tudattalanul" történik.

3. Megértés és megnyilatkozás

A nyelvi modul felépítésének az az alapelve, hogy a megértés és a mondatlétrehozás (szemben a Chomsky-féle „genetikus" elmélettel) csak intelligencia segítségével lehet sikeres. Lehet, hogy az ismert szavak felismerése és jelentésüknek a memóriából való előkeresése (és az ellenkező folyamat, a szerkezetek kimondása) automatikus, tudattalan folyamat, amelyet kellő szenzomotoros mechanizmusok segítségével akár kutyák vagy majmok is végre tudnának hajtani. Ezekre az amúgy nagyon fontos folyamatokra összpontosítanak a mai számítógépes nyelvészet főáramába tartozó statisztikaalapú rendszerek, de éppen ezekről a tudásalapú rendszereknek nem sok mondanivalójuk van. Nagyrészt az intelligens szférához tartozik azonban az, ahogyan kisakkozzuk, hogy beszélgetőtársunk miért éppen egy bizonyos nyelvtani vázba miért éppen azokat a szavakat illesztette bele, amelyeket hallunk, illetve hogy milyen vázba milyen szavakat kell illesztenünk, hogy a hallgatóságunkban egy bizonyos jelentésélményt idézzünk fel.

Ezt a folyamatot a logikában **abdukciónak** nevezett okoskodási eljárással modelláljuk. Az abdukció azt jelenti, hogy egy következtetésnek a konklúzió-

ját ismerjük, és olyan premisszákat keresünk, amelyekből következik. Persze sok premisszahalmaz eleget tehet ennek, természetesen az adott körülmények között legvalószínűbbet keressük ezek közül a halmazok közül. A jelen esetben a biztos információt, az elhangzott szavak sorát konklúziónak tekintjük, és legjobb tudásunk szerint kiegészítjük egy koherens mondatábrázolássá, szőröstül-bőröstül, vagyis a mondat jelentésével együtt, olyannal, amelyben a forma harmóniában van a tartalommal (vö. [3]). Ennek során a legkülönbözőbb információkra szükségünk lehet, nemcsak nyelvtaniakra, és — hacsak nem nagyon olajozott beszélgetésről, rutinszerű nyelvhasználatról van szó — nemcsak olyanokra, amelyek automatikusan, tudattalanul hozzáférhetőek.

4. Konstrukciós nyelvtan

A tudásalapú nyelvfeldolgozás egyik kulcsmozzanata, hogy a nyelvi tudást nem procedurális formában kell tárolnunk, és nem szabad modularizálnunk, vagyis nem különíthetjük el élesen az alaktani, mondattani, jelentéstani és pragmatikai tudást. Ennek a problémának a kézenfekvő megoldása az ún. **konstrukciós nyelvtan** (ld. pl. [4]) használata.

A konstrukciós nyelvtan legfőbb elve ugyanis az, hogy az embereknek nem azt a képességét kell megmagyarázni, hogy különbséget tudnak tenni helyes és helytelen mondatok között (már az is kérdéses, hogy egyáltalán rendelkeznek-e ezzel a képességgel), hanem azt, hogy bizonyos jelentéstípusok és formatípusok közötti kapcsolatok ismeretében képesek egymással kommunikálni.

A konstrukciós nyelvtanban egyetlen formája van a nyelvi szabályszerűségek leírásának, mégpedig a **konstrukció**, amely nem más, mint egy általánosítás formai és jelentéstani tulajdonságok konvencionális együttjárásáról. (Speciális konstrukciókként felvehetőek még csak formai vagy csak tartalmi általánosítások is, valamint konstrukciók egymással való **asszociációjáról**, konvencionális együttjárásáról is lehet nyilatkozni, sőt, akár arról is, hogy két konstrukció taszítja, gátolja egymást.) A nyelvtan nem más, mint rengeteg tartalmi és formai általánosítás együttjárásának és kölcsönös gátlásának bonyolult szövedéke.

Mind a mondatelemzés, mind a mondatlétrehozás folyamatában voltaképpen szükséges konstrukciók aktivizálásával párhuzamosan abdukciós folyamat, vagyis az aktivizált konstrukciók koherens egésszé való kiegészítése történik — ehhez általában újabb konstrukciók aktivizálására van szükség.

4.1. Formai általánosítások

A formai jellemzések fonológiai és mondattani mintákról szólnak. A kettő elég különböző, és különböző automatikus folyamatok működnek rajtuk, de nem annyira, hogy ne lehessen egyetlen leíró nyelv segítségével beszélni róluk. Mindkettőben a szemantikából ismert **eseményszerkezetet** írunk le, amelyben több szálon futnak az események, és az eseményfonalak egy-egy pontja egymáshoz képest időzítve (szinkronizálva) van (vö. [1]). A fonológiában jól ismertek ezek

az eseményfonalak, többé-kevésbé független artikulációs-akusztikai mechanizmusoknak felelnek meg (ezek az autoszegmentális fonológia „tier"-jei, tengelyei). A mondattanban is vannak ilyen fonalak, például a predikátum—argumentum szerkezet, a diskurzusbeli szerepeknek megfelelő szerkezet stb. Automatikus moduljaink arra valók, hogy az ilyen eseményszerkezetet leíró nyelven vagy nyelveken villámgyors összegzéseket és konzisztencia-ellenőrzéseket végezzenek.

4.2. Jelentéstani jellemzések

Sokkal nagyobb gondok vannak a jelentéstani jellemzésekkel a konstrukciós nyelvtanban. Szemben a frázisstruktúra-nyelvtan rendezett, fegyelmezett eljárásával, ahol minden ponton a szánkba rágják, hogyan kell a jelentéseket összerakni, itt sok helyről, sok párhuzamosan futó szálon sokféle információ gyűlik, és ezeket kell összegezni.

Ráadásul nemcsak a struktúra lazasága, pontosabban szerteágazó volta nehezíti meg a jelentéstani jellemzést, hanem az a tény is, amit már említettem, hogy a „nyelvtani szempontból fontos" jelentéstani jellemzés nem különíthető el élesen egyéb, a jelekhez kapcsolódó információktól, az utóbbiak tehát folyton „feltolulnak", ezt is menedzselni kell. És végül, habként a torta tetejére, ott van az a már-már filozófiai probléma, hogy mit is jelent egy mondatot megérteni. Felületesen „megértjük" azt a fél mondatot is, amit a buszon véletlenül meghallunk, tehát nem korlátozhatjuk a megértés fogalmát arra, amikor valóságosan felismerjük a beszélő kommunikációs szándékát. De ha csak az „igazi" kommunikációs helyzeteket nézzük, akkor is nagyon változó, hogy milyen mélyen derítjük fel, ami a kimondottak mögött van. Lakonikusan csak annyit tudunk mondani erről, hogy „megértésről" értelmesen csak úgy lehet egyáltalán beszélni, ha a hallgatóság céljaihoz képest értjük ezt, vagyis akkor értünk meg valamit, ha választ kapunk valamire, amire célunk volt választ kapni.

Nos, mindezek a bonyodalmak olyan jelentéstani ábrázolást tesznek szükségessé, amely radikálisan különbözik a szokásostól. A hagyományos formális szemantikában ugyanis minden mondattani kategóriához a megfelelő típusú jelentés tartozik, ezért tudjuk olajozottan összeépíteni őket, ha megfelelő mondattani szerkezetben találkozunk velük. Tehát a „hiányos" mondatszerkezethez (pl. alany nélküli mondat) „hiányos" jelentés (azaz funktor, operandus nélkül hiányos kifejezés) tartozik, a mondattani predikátum—vonzat viszony a jelentéstanban funktor—operandus viszonynak felel meg, és így tovább.

Mivel az általunk feltételezett mondatszerkezet nem szigorúan hierarchikus, másfajta értelmezési módszert, másfajta jelentéstani szemléletet kell választanunk. Ugyanakkor a kompozicionalitás elvét meg akarjuk tartani, tehát az értelmezésnek a mondatszerkezettel összhangban kell történnie, tehát a mondatszerkezetnek összhangban kell lennie azzal, ahogyan a részek jelentését kombináljuk. De a kombinációt nem „vezérli" a mondattan abban az értelemben, ahogy a hagyományos generatív nyelvtanban, vagyis nem áll fenn megfelelés a mondattani kategóriák és szemantikai típusok között. Sőt, az igazság az, hogy szokásos értelemben vett mondattani kategóriákról és szemantikai típusokról nem is beszélhetünk.

Végül a megértés mélységének filozófiai problémájáról annyit, hogy — akárhogy oldjuk is meg ezt a problémát — csak akkor tudunk megfelelni a „változó mélységű megértés" követelményének, ha az egész megértési folyamat **fokozatos finomítás** formáját ölti. Ez pedig azt jelenti, hogy a „megértés mint (logikai nyelvre való) fordítás" elképzelésének véglegesen búcsút mondhatunk. Inkább úgy kell felfogni a dolgot, ahogy az abdukcióval kapcsolatban is mondtam, hogy iteratív módon kell hipotéziseket megfogalmazni a hallottak elhangzásának egyre mélyebb és mélyebb lehetséges okairól, mindaddig, amíg el nem jutunk egy olyanhoz, amely választ ad arra, amit tudni akartunk (persze, csak ha van rá időnk).

5. Tanulás

A rendszer továbbfejlesztésében messze a legnagyobb problémát az jelenti, hogy tudásalapú rendszerek felépítését automatikus tanulással igen nehéz szimulálni. Míg a hangok és hangtípusok száma igen korlátozott, de még a szótípusoké is meglehetősen, a jelentéstípusok lényegében korlátlan számúak, és reménytelennek olyan korpuszt összeállítani, amelyben minden forma—jelentés társításra elegendő mennyiségű példát lehetne találni. Nem véletlen, hogy ehhez a valóságos életben is évek megfeszített munkájára van szükségünk. Ennek ellenére biztosak vagyunk abban, hogy a rendszer feltöltése nem történhet kizárólag „kézi" erővel. Mint a természetes nyelv feldolgozásában általában, a tanulás esetében is az az álláspontunk, hogy csak a szakértői tudás és az automatikus tanulási módszerek kombinálása, tehát egy **hibrid** megoldás lehet eredményes.

Hivatkozások

1. Bird, Steven, és Ewan Klein: *Phonological Events*. EUCCS/RP-24, Centre for Cognitive Science, University of Edinburgh, Edinburgh, 1989.
2. De Smedt, Koenraad, Helmut Horacek és Michael Zock: *Trends in Natural Language Generation: An Artificial Intelligence Perspective*. Springer-Verlag, 1996. 'Architectures for natural language generation: Problems and perspectives', pp. 17–46.
3. Hobbs, Jerry R., Mark Stickel, Douglas Appelt és Paul Martin: 'Interpretation as abduction'. *Artificial Intelligence* **63** (1993), 69–142.
4. Kálmán László: *Konstrukciós nyelvtan*. Tinta Könyvkiadó, Budapest, 2001.

Knowledge-Based Natural-Language Processing

László Kálmán[1], László Balázs[2], and Miklós Erdélyi Szabó[3]

[1] Applied Logic Lab, Budapest
Research Institute for Linguistics, Hungarian Academy of Sciences, Budapest
Theoretical Linguistics Programme, Loránd Eötvös University, Budapest
kalman@nytud.hu
[2] Applied Logic Lab, Budapest
bazsi@all.hu
[3] Applied Logic Lab, Budapest
Alfréd Rényi Institute for Mathematics, Hungarian Academy of Sciences, Budapest
mszabo@renyi.hu

Keywords: computational linguistics, syntax, semantics, construction grammar, knowledge-based approaches

Computational linguistics' enthusiasm for the formal models of syntax and semantics introduced by Chomsky and Montague in the 70's declined during the 80's and 90's. It turned out that an autonomous syntax, without access to semantics, or an autonomous semantics, without access to pragmatics and world knowledge, are hopelessly untractable computationally and not suitable for applications.

The alternative that we propose is that modules are needed, but they must correspond to different types of linguistic knowledge and different modes of operation (such as statistical, associative and logical modes) rather than different linguistic 'levels'. The statistical modules perform jobs like speech recognition and synthesis, associative modules take care of invoking forms, structures and meanings from memory, and logical modules perform reasoning in understanding and planning tasks.

The main ingredient of our system is **abduction**: both understanding and producing utterances requires us to complete an array of pieces of information about the formal and semantic aspects of an utterance. All sources of information, including linguistic, pragmatic and cognitive sources can be used in order to achieve this goal. This is the key procedure that our system performs.

Linguistic knowledge itself is stored in a format that does not draw a sharp line between linguistic, pragmatic and cognitive aspects of signs. The framework that makes this possible is called **construction grammar**. Construction grammar is about systematic (conventional) associations of form types and meaning types, and information about the probabilities of their co-occurrence. The starting-point of the abduction process in our system is a loosely connected network of 'activized' constructions.

Szemantikai hálószótár diszfáziaterápiához

Bácsi János

Bacsi@jgytf.u-szeged.hu

Absztrakt: A megkésett beszédfejlődésű (diszfáziás) gyermekek beszédterápiája eredményesebben tervezhető, végezhető számítógéppel segített tanítási – tanulási programok alkalmazásával, mint a hagyományos, logopédusok által alkalmazott ún. hívóképes módszer segítségével. Célunk egy olyan szemantikai hálószótár létrehozása – nagy mennyiségű empirikus adatbázis alapján-, amely megjósolja, hogy egy fogalomhoz milyen más fogalmak kapcsolódnak a 4-7 éves gyermekek RTM-jében, vagyis egy már meglévő fogalomhoz milyen más fogalmat tud a terapeuta a legkönnyebben kapcsolni. A szemantikai hálószótár ezen felül alkalmas lesz arra, hogy a már olvasni tudó diák a hálószótár kapcsolt szavain egy virtuális sétát tudjon tenni pusztán mikrofonba bemondott szavak alkalmazásával. A szoftver ezáltal segíti a gyermeket, hogy az így bejárt szemantikai struktúrák megerősödjenek vagy kiépüljenek.

Kulcsszavak: megkésett beszédfejlődés, gyakorisági szótár, hálóelméletek, szemantikai struktúra, számítógéppel segített tanulás

Bevezetés

A kognitív tudományok, a memóriakutatás, a mesterséges intelligencia kutatása, a modern nyelvészet, a számítógépes nyelvészet, az evolúciós pszichológia és a modern tanuláselméletek olyan kutatási eredményeket mutattak fel, amelyek arra ösztönözték a diszfáziás gyermekekkel foglalkozó terapeutákat valamint az írott nyelv elsajátításával foglalkozó kutatókat, hogy eddig alkalmazott módszereiket vizsgálják felül, és gondolják újra.

Az expresszív nyelvi zavarral küzdő gyermekek fejlesztése nem hanyagolható el, mivel a diszfáziás nyelvi fejlődés nem nyelvspecifikus, a világ valamennyi nyelvében – ahol végeztek ilyen felmérést – az adott populáció (4-8 évesek) 4 – 7 %-át érinti. Az érintett gyermekekkel azért kell differenciáltan foglalkozni, mert speciális fejlesztés nélkül számukra problémát okoz a vizuális nyelv (írás-olvasás) elsajátítása. A diszfáziás nyelvi fejlődésű gyermek az intézményesített oktatás folyamatában diszlexia és/vagy diszgráfia tüneteit fogja produkálni, amit nem az iskolából kap, mint valami fertőzést, hanem az expresszív nyelvi deficit miatt a vizuális nyelv elsajátításához szükséges készségek nem megfelelő szintje hoz elő.

Az expresszív nyelvi zavar legfontosabb jellemzője a limitált szókincs (DSM-IV, 1995), ezért a fejlesztést tervező és végző terapeuta legfontosabb feladata a szókincsbővítés. Ezt a munkát általában logopédusok végzik, akik a

szókincsfejlesztéshez azokat a hívóképeket használják, amelyek az olvasástanításhoz használatosak. A hívóképekkel az a probléma, hogy egy négyéves diszfáziás gyermek meg sem tudja nevezni azt, hogy mit lát a képen, vagy ha igen, a 40 kép között nem talál semmilyen szemantikai kapcsolatot, így a rendszeres ismétlés hatására hosszú idő alatt úgy tanulja meg azokat, mint „diszkrét egységeket".

A szemantikai hálószótár létrehozásával az a célunk, hogy a nagy mennyiségű empirikus adat alapján láthatóvá tegyük, hogy a 4-8 éves gyermekek RTM-jében egy-egy inputként jelentkező fogalomhoz milyen más fogalom kapcsolódik a legkönnyebben, vagyis hogyan jön létre az RTM-ben egy szemantikai struktúra. A szemantikastruktúrák súlyozott kapcsolatai megjósolhatóvá teszik, hogy a diszfáziás gyermek limitált szókincséhez milyen szavak kapcsolhatók a legkönnyebben és a leggyorsabban, vagyis milyen új terápiás lehetőséget nyújt a szemantikai hálószótár.

Dolgozatomban azt mutatom be, hogyan készül a szótár, mik az eddig elért eredményeink, milyen feladatot kell még megoldani.

I. A szemantikai struktúra

Az emlékezetkutatás (Baddeley, 2001) bizonyította, hogy könnyebben tanuljuk meg azokat a fogalmakat, amelyekhez valamilyen előismeretet tudunk kapcsolni, mint azokat, amelyekhez nem. Egy új fogalom megtanulásának egy mozzanata úgy mehet végbe, hogy a rövid távú memóriába (RTM) bekerülő új fogalom a hosszú távú memóriából (HTM) előhív 1-6 olyan fogalmat (előismeretet), amelyet valamilyen kapcsolatba tud hozni az új fogalommal. (Ha az új fogalom semmit nem hív elő a HTM-ből, akkor vagy törlődik, vagy egy „közvetítő fogalmat" kell keresni, amely az új fogalmat kapcsolatba tudja hozni egy a HTM-ben tárolt fogalommal, így e hármas kapcsolódás segíti, létrehozza a szótanulást.)

Az RTM-ben létrejövő kapcsolatokat (akár külső, akár belső hívóinger hatására jön létre) szemantikai struktúrának nevezem, amely alatt egy olyan fogalmi hálót értek, amely az RTM-ben jön létre úgy, hogy az oda bekerülő hívóinger a HTM-ből előhív 1-6 olyan fogalmat, amely össze tud kapcsolódni a hívóinger fogalmával. (Bácsi, 2003) (A szemantikai struktúra azért áll 2-7 fogalom kapcsolatából, mert minimum 2 fogalom kapcsolata jelöl ki egy harmadikat, a felső korlátra (7 fogalom) pedig azért van szükség, mert az RTM kapacitása nem tud többet kezelni (Miller, 1996). Eddigi vizsgálataink azt mutatják, hogy a 6 évesek RTM kapacitása átlag 4-5 egység.)

Ha empirikus vizsgálattal feltérképezzük, hogy az egészséges 5-8 éves gyermekek RTM-jében milyen szemantikai struktúra jön létre egy hívóinger hatására, akkor a feldolgozott empirikus anyag alapján ki fog derülni, hogy milyen egységekből áll egy szemantikai struktúra, s ez alapján, ennek felhasználásával hatásosabb, eredményesebb (gyorsabb és rövidebb) terápiákat tudunk tervezni, a diszfáziás gyermekek fejlesztéséhez, mint az eddigiek.

II. A szemantikai hálószótár készítésének menete

II.1. A 6 évesek szóhasználatának gyakorisági szótára

Ahhoz, hogy meg tudjuk állapítani, melyek a már biztosan kialakult szemantikai struktúrák az érintett populáció esetén, készítenünk kellett egy gyakorisági szótárat, amely relevánsan tükrözi, hogy az érintett korosztály mely szavakat használja és/vagy mely szavakkal találkozik a leggyakrabban (Bácsi-Kerekes, 2003).

A szótár elkészítéséhez az OM által engedélyezett 13 elsős olvasókönyv teljes szóanyagát feldolgoztuk. Úgy gondoltuk, hogy az olvasókönyvek szóanyagából készített gyakorisági szótár relevánsan tükrözni fogja, melyek azok a leggyakrabban előforduló szavak, amelyekkel egy hatéves biztosan találkozik a vizuális nyelv elsajátításának folyamatában.

A tankönyvek teljes írásos anyagának feldolgozása után először töröltük azokat a grafémasorokat, amelyek nem elemei anyanyelvünk szókészletének, de az olvasástanítás menetében nélkülözhetetlen hangkapcsolatok. Ezek után töröltük a nem tartalmas szavakat. (Tartalmas szónak azt tekintettük, amely a jelentésnek mindhárom oldalát hordozza: (lexikai, grammatikai és pragmatikai)

A szótár néhány mennyiségi mutatóját az 1. ábra szemlélteti.

Grafémasorok száma	27.297
Tartalmas szavak száma	12.226
10 vagy annál gyakrabban előforduló szavak száma	1.953

1. ábra

A szemantikai hálószótár létrehozásához a 200 leggyakrabban előforduló köznevet használjuk fel. (Azért a közneveket, mert tanulásuk elsődleges a többi szófajhoz viszonyítva. Az igéket is fel fogjuk dolgozni, de más aspektusból.)

II.2. Anyaggyűjtés

A szemantikai hálószótárhoz az anyaggyűjtést óvónők, tanítók és főiskolai hallgatók végzik jelenleg az ország 20 különböző településén. Az anyaggyűjtés alanyai az óvodák nagycsoportosai, valamint az általános iskolák 1. és 2. osztályos tanulói. Azt szeretnénk, hogy a 200 leggyakoribb főnév mindegyikéről ötezer gyermek mondja el, hogy mi jut az adott szóról eszébe. Szóbemondólapokat készítettünk, amelyeken öt-öt szó szerepel, pl.: apa, anya, ember, fa, szó. A vizsgálat vezetője felteszi a következő kérdést: „Mitől fa egy fa?" Ha a gyermek már tudja a vizuális nyelvet, akkor írásban válaszol, ha még nem, akkor szóban, és a vizsgálat vezetője jegyzi le válaszait. Mennyiségi korlátozást nem szabunk, minden feladat a következő felszólítással kezdődik: „Mondd meg .../ Írd le azt a néhány fogalmat...!"

Az eddig gyűjtött és feldolgozott eredmények azt mutatják, hogy a gyermekek egy hívószóra átlag 2,8 szóval vagy szókapcsolattal válaszolnak, amiből az a következtetés vonható le, hogy kb. 2.800.000 szó feldolgozása alapján készül el a szemantikai hálószótár. Természetesen ezek nem különböző szavak. Az eddigi

feldolgozás alapján azt tudjuk elmondani, hogy kb. 350 különböző fogalom kapcsolódik egy hívószóhoz 5000 gyermek bemondása alapján. (Az ingadozás igen nagy: az anya hívószó esetén 741 különböző asszociációt kaptunk, a szó esetén pedig csak 178-at, eddig ez a két véglet.) Az egy-egy hívószóhoz kapcsolódó fogalmak esetén pedig szintén lesznek átfedések, ezért még nem tudjuk megmondani, hány fogalmat tartalmaz majd a szemantikai hálószótár.

A következőkben öt olyan szót mutatok be, amelyek feldolgozását az adott korpuszon befejeztük: (a számok zárójelben azt jelzik, mennyi az adott szóhoz társuló különböző asszociók száma):

anya: *szeret* (1588), *szülő* (1508), *kedves* (1263), *szeret* (1127), *szeretem* (875);
apa: *szülő* (1213), *dolgozik* (920), *szeretet* (874,) *szeret* (825), *kedves* (623);
ember: *élőlény* (1813), *élet* (1083), *él* (1046), *fej* (934), *okos* (930);
fa: *növény* (2175), *élőlény* (994), *levél (711), ág* (587), *levegő* (514);
szó: *beszéd* (908), *betű* (875), *mondat* (787), *hang* (772), *beszél* (685).

II.3. Hálóelméletek

Szótárunk létrehozásához áttanulmányoztuk a szemantikai hálóelméleteket, valamint azok kritikáit (Quillian, 1969; Collins és Loftus, 1975; Janson – Laird – Hermann – Chaffin, 1984) valamint a thesaurusokat.

A thesaurusokkal az a probléma, hogy nem adnak semmiféle információt az egyes lexémák jelentésének összefüggéseiről, valamint a különböző nyelvváltozatokból származó (regionális, szociális, szakmai stb.) lexémákat minden megjegyzés nélkül hozzák egymással kapcsolatba. (D. Crystál, 1998)

A hálóelméletek és a thesaurusok kritikái mutatták meg számomra, hogy mégis miért éppen ezekhez az elméletekhez hasonló hálómodell lehet a legalkalmasabb elméleti keret a hatékonyabb terápiák kidolgozásához. A kritikusoknak igazuk van, amikor azt mondják, hogy a hálómodellek túl erőteljesek, hogy szinte csak a fogalmak közötti kapcsolatok létrejötte az érdekes, de nem tudnak számot adni a fogalom és a világ kapcsolatáról, ami pedig a szemantika alapfeladata.

Az eredményesebb terápia szempontjából viszont éppen az lehet a fontos, hogy az adott populációra nézve a fogalmak között milyen kapcsolatok vannak már meg, s a nagymennyiségű empirikus adat ezt képes szemléltetni. Képes megmutatni, hogy egy adott fogalomhoz a gyakorisági mutatók alapján milyen más fogalmak társíthatók a legkönnyebben, a legeredményesebben, s hogy e már társított fogalmakhoz milyen más fogalmak stb.

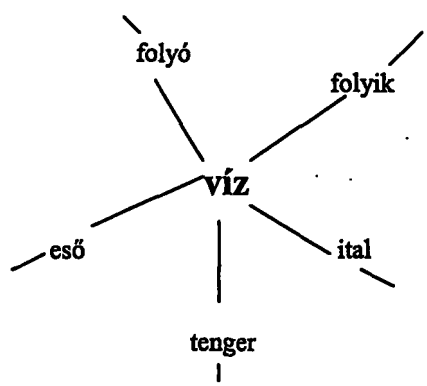
Bár a szemantikai struktúrák szerveződésében nem egy absztrakt szemantikai szabály érvényesül, amely bármit is mondani tudna egy szó lexikai jelentéséről, hanem valamiféle szubjektív tudás, mégis az empirikusan összegzett szubjektív tudások összessége talán mond valamit a szavak jelentésszerveződéséről, és nemcsak a releváns szavakhoz rendelt szabad asszociációt prezentálja, hanem feltár olyan kapcsolatot a lexémák között, amelyet minden ember létrehoz, vagyis tükrözi a kollokációt.

Szemantikai hálószótárunk azokat a kollokációkat fogja megmutatni, amilyet Londman interszubjektív világtudásnak nevez (Landman, 1982), így nemcsak a

fogalmak kapcsolatáról, hanem a kapcsolatok által kijelölt új fogalmakról is számot fog adni.

III. A szoftver működése

Ha a teljes szóanyagot feldolgoztuk, az informatikusok olyan programot fognak írni, amely súlyozottan a háló elemei között az összes összeköttetést prezentálja. Ha beírunk vagy bemondunk a számítógépbe egy szót, amely a szótár eleme, akkor az a lexéma megjelenik a képernyő közepén, és súlyozottan kapcsolva az a másik öt szó is, amely az empirikus anyag feldolgozása alapján a leggyakoribb kapcsolatban volt az adott szóval. Ha a beírt szóval szemantikai struktúrát alkotó öt szó közül bármelyikre ráklikkelünk vagy bemondjuk, akkor az jelenik meg a képernyő közepén hívószóként a hozzá kapcsolódó öt leggyakoribb fogalommal együtt. Mivel a szoftver minden lehetséges összeköttetést tartalmaz, ezért a teljes szókészleten végig lehet járni úgy, hogy mindig az adott szemantikai struktúra jelenik meg a képernyőn. Az eddig elmondottakat a 2. ábra sematikusan szemlélteti a víz hívószó kapcsán.



2. ábra

A szótárat ellátjuk egy valós idejű beszédhang-felismerő rendszerrel, ami lehetővé teszi azt, hogy a már olvasni tudó tanuló pusztán mikrofonba bemondott szavak alkalmazásával egy „virtuális sétát" tudjon tenni a hálószótár kapcsolt szavain. Ez a „séta" pedig segít abban, hogy a bejárt szemantikai struktúrák megerősödjenek vagy kiépüljenek a gyermek memóriájában, azaz segíti a szótanulást.

Rendelkezünk egy valós idejű beszédhang-felismerőrendszerrel. Kutatócsoportunk a már említett első osztályos olvasókönyvek alapján készült gyakorisági szótár 1953 szava alapján (ennyi szó fordult elő legalább 10 vagy annál nagyobb gyakorisággal) az ország 11 gyakorlóiskolájában 500 adatközlőtől (6-7 éves tanulók) 100-100 szószintű bemondást rögzített számítógépen. A rögzített szavakat szegmentáltuk és annotáltuk. Ez a hangadatbázis képzi a BeszédMester szoftverünk alapját (Paczolay,

Tóth, Kocsor, Kerekes, 2003), amely a beszédterápiában és az olvasási készség fejlesztésében használható szoftver.

A közel 250.000 szegmentált és annotált beszédhang, valamint az 50.000 rögzített szó (amely tartalmazza a szemantikai hálószótár alapját képező 200 leggyakoribb főnevet) fogja a szótárunk beszédhang-felismerő moduljának alapját képezni.

Virtuális szemantikai hálószótárunktól azt várjuk, hogy segítse a terapeutákat az expresszív nyelvi zavarral küzdő gyermekek fejlesztését szolgáló tervek elkészítésében, valamint nyújtson segítséget az egészséges gyermekek szókincsfejlesztésében is.

Összegzés

A számítógéppel segített tanulás – bármi legyen a tanulás tárgya – óriási motivációs erővel bír. Napjainkban a gyermekek hamarabb tanulják meg kezelni a számítógépet, mint az olvasást vagy az írást.

A számítógéppel segített tanulás olyan didaktikákat, olyan programokat kíván, amelyek eddig nem voltak jelen az intézményesített oktatásban és a beszédterápiákban. A megfelelő didaktika keresése, a programok létrehozása új vonatkoztatási keretet adhat egy-egy tudományos vizsgálatnak. A számítógéppel segített oktatás azért módosít az adott tudomány(ok) vizsgálati keretein, hogy a saját kutatási céljaira tudja alkalmazni azt. Ezt tettük mi is, amikor újragondoltuk a szemantikai hálóelméleteket, hiszen a célunk nem egy új szemantikaelmélet létrehozása, hanem a pedagógia gyakorlatában hasznosítható tervek megalkotása. Hogy létezik-e szemantikai struktúra, hogy tudományos-e az asszociációs háló vizsgálata, nem a mi alapkérdésünk. A gyakorlatból viszont meggyőző bizonyítékaink vannak arra, hogy a szemantikai struktúra meglétének, kiépülésének, tudatos kiépíthetőségének feltételezése hasznos és eredményes a beszédterápia és a szókincsfejlesztés metodikájában.

Hogy feltételezéseinknek a gyakorlati hasznon túl lehet valami tudományos alapja, azt bizonyíthatja az, hogy Pléh az asszociáció reneszánszáról beszél a kognitív pszichológiában (Pléh, 2003), és azt javasolja: „...az asszociáció és a struktúra vagy a logika egyaránt használandó a magasabbrendű folyamatok modellálásában."

Irodalom

1. Bácsi János – Kerekes Judit (2003) Az első osztályos olvasókönyvek szóanyagából készült gyakorisági szótár "Van szó". Módszertani Közlemények. 2003. II. szám 52-57.
2. Bácsi János (2003): A hálószemantika szerepe a megkésett beszédfejlődés terápiájában. Alkalmazott Nyelvészeti Konferencia Füzetei. 274-283.
3. Baddeley, A. (2001) Az emberi emlékezet. Budapest: Osiris Kiadó

4. Collins, A. M. – Loftus, G. R. (1975) A Spreading activation theoriy of semantic processing. Psycholoqical Review, 82. 240-247.
5. Crystal, D. (1998): A nyelv enciklopédiája. Budapest, Osiris Kiadó.
6. Jonson-Laird, P. N. – Herrmann, D. J. – Chaffin (1984) Only connections: A critique of semantic networks. Psychological Bulletin, 96, 292-315.
7. Kerekes Judit (2003): BeszédMester. Alkalmazott Nyelvészeti Konferencia Füzetei. 44-49.
8. Landman (1982:) Varieties of formal semantics. Proceedings of the 4. Amsterdam
9. Miller, G. A. (1956) The maqical number seven, plus or minus two: Same limits on our capacity for processing information. Psychological Review, 63. 81-97.
10. Paczolay Dénes, Tóth László, Kocsor András és Kerekes Judit (2002) Gépi tanulás alkalmazása egy fonológiai tudatosság – fejlesztő rendszerben. Alkalmazott Nyelvtudomány II. évfolyam 2. szám 55-67.
11. Pléh Csaba (2003): Az asszociáció reneszánsza a kognitív pszichológiában. http://sol.cc.u-szeged.hu~pleh/magyar/cikkek/regi/4asszren.html
12. Quillian, M. R. (1969) The teachable lanquaqe comprehender: A simulation program and theory of lanquaqe. Communication of the ACM, 12. 459-476.

A Semantic Network Dictionary for Dysphasia Therapy

Bácsi János

bacsi@jgytf.u-szeged.hu

Keywords: delayed speech development, frequency dictionary, network theories, semantic structure, machine-aided speech therapy

The purpose of this project is to compile a network dictionary running on computers to help the conscious planning of the speech therapy of 4–8 year-old-children suffering from expressive language disorder. The children will be able to take a virtual tour through the connected words of the network dictionary only by saying words into a microphone, by which the program will help in the development and maintenance of semantic structures.

Our basic psychological assumption, reinforced by our test results, is that a call word that enters the STM (short-term memory) will retrieve about 2–5 concepts from the LTM (long-term memory) and its association with them produces the semantic structure of the call word.

One of our linguistic achievements is that we have compiled a frequency dictionary based on thirteen primers. It contains 27,293 grapheme sequences of which 12,226 present content words. The goal of the dictionary is to find out which are the words that a six-year-old child are most likely to come across during learning the written language. The empirical database of the network dictionary will be drawn by asking 5000 children aged 4–7 to tell us what comes into their minds when they hear the 200 most frequently used nouns of the frequency dictionary. We have done half of that work so far. The five more frequent associations prompted by the word *anya* 'mother,' which is among the words that have been completely processed, are *szeret* 'she loves me' 1588, *szülő* 'parent' 1501, *szeretet* 'love' (noun) 1127, *szeretem* 'I love her' 875. The number of the distinct associations prompted by 'mother' is 741. We also have a large database of segmented and annotated recordings of children's voice, which contains 250,000 items.

The computational task to produce a program for the network dictionary that uses speech recognition has already been accomplished. The other task is to create a network dictionary that demonstrates all of the possible associations based on the empirical material that has accumulated 400,000 words so far.

The expected result makes it predictable what are the most like associations evoked by certain concepts among children aged 4–7, which makes the therapy of delayed speech development plannable using the words that the children have already learnt.

A természetes nyelvek formális modelljeiről

Prószéky Gábor

MorphoLogic
Budapest
proszeky@morphologic.hu

1 Bevezetés

A kérdés, hogy mitől „természetes" egy nyelv, általában úgy válaszolható meg, hogy: a természetes nyelvek azok, amelyeket nem a nyelvleíró definiál, hanem adottnak tekinthetők, így a kutatónak egyetlen lehetőség marad: hogy nyelvtanokat – az eredeti objektumot formális eszközökkel jellemző rendszereket – hozzon hozzájuk létre. A formális nyelvészet 1950-es évekbeli kialakulásával megjelent a kérdés: van-e formálisan kezelhető nyelvtana a természetes nyelveknek, illetve hogy matematikailag egyáltalán formalizálhatók-e ezek a nyelvtanok? A ravaszabb kérdés persze az, hogy az ilyen modell-nyelvtanok által generált nyelvek tényleg a kiinduláshoz használt természetes nyelvek-e? Egy bizonyos: a természetes nyelvek tipikus osztályzásai mind a mai napig elsősorban aszerint történnek, hogy a természetes nyelvek leírásához készített nyelvtanok ún. gyenge generatív kapacitása milyen. Ez más szavakkal azt jelenti, hogy a különböző nyelvtanok által generált füzérhalmazok összehasonlítása melyik nyelvtant hozza ki „győztesnek".

A bonyolultságelmélet a nyelvi modellek világának megismeréséhez is hozzájárult. A különféle modellekben a nyelvtani jelenségek számítógépes kezeléséhez szükséges idő- és helyigényeket ki tudjuk mutatni a segítségével. A bonyolultságelmélet segítségével lényegesen finomabban tudjuk kezelni az előzőleg csak a Chomsky-hierarchia által definiált néhány osztályba sorolt természetes nyelvi grammatikamodelleket [1][2]. A gyenge generatív kapacitáson alapuló osztályzás alapján sokan a feldolgozási bonyolultságra is következtetni véltek. A bonyolultságelmélettel kimutatható az egyes leírható formalizmusokban megbúvó nem várt komplexitás is, vagyis az, hogy egy Chomsky-hierarchiában egyszerűbbnek definiált gépezet nem feltétlen tudja garantálni a feldolgozásbeli hatékonyságot. Arról nem is beszélve, hogy ha már tudjuk, hogy mi okozza a bonyolultságot, lehet esélyünk arra is, hogy rájöjjünk, miképpen lehetne formalizmusunkat egyszerűbbé tenni.

2 Az elmúlt 50 év grammatikai formalizmusairól

Chomsky híres, *Syntactic Structures*-beli definíciója a természetes nyelvről így hangzik [3]: *„Nyelvnek tekintem a mondatok valamely (véges vagy végtelen) halmazát; minden egyes mondat véges hosszúságú és elemek véges halmazából épül fel. [...] Valamely Ny nyelv nyelvészeti elemzésének alapvető célja az, hogy a nyelvtanilag helyes sorozatokat, amelyek Ny mondatai, különválasszuk a nyelvtanilag helytelen sorozatoktól, amelyek Ny-nek nem mondatai. [...] Ny nyelvtana ily módon olyan készülék lesz, amely Ny valamennyi nyelvtanilag helyes sorozatát létrehozza, azaz generálja, de nem generál egyetlen nyelvtanilag helytelent sem."* Ez a definíció az alábbi állításokra bontható szét: (1) Egy természetes nyelv valamifajta objektumok (ezeket szokás mondatoknak nevezni) egyfajta összessége. (2) Ez az összesség halmaz. (3) Minden mondat véges objektum. (4) A természetes nyelveknek véges építőelem-halmazuk (szótáruk) van. (5) Minden *Ny* nyelv leírható

olyan eszközzel, mely felsorolja *Ny*-et. A természetes nyelvek esetében ezt a (véges) eszközt az *Ny* természetes nyelv nyelvtanának nevezzük.

Maga Chomsky a halmaz elemein operáló újraíró szabályok alakjára vonatkozó megkötések alapján létrehozta híres nyelvosztályait: a megszorítás nélküli, a környezetfüggő, a környezetfüggetlen és a reguláris nyelvek kategóriáit. A reguláris jellemzésről még ugyanebben a művében kimutatta, hogy nem felel meg a természetes nyelvek leírására. A bizonyításhoz leggyakrabban használt önbeágyazás olyan természetes nyelvi jelenség ugyanis, mely magában hordja a reguláris nyelvtannal való jellemezhetetlenséget. Már csak az a kérdés, hogy a formális nyelvekben egyszerűen bemutatható jelenség valóban megtalálható-e – és ha igen, milyen mértékben – a természetes nyelvekben? Ilyenkor természetesen a nyelv modelljének regularitásáról (és nem a nyelv regularitásáról) beszélünk. A formális nyelvek világában könnyen találunk példát az önbeágyazásra: az $S \rightarrow aSb$ és az $S \rightarrow \epsilon$ szabályokból álló nyelvtan önbeágyazó módon hozza létre az $L=a^nb^n$ formális nyelvet. Ugyanakkor a természetes nyelvben – mondjuk a magyarban – ugyanennek a jelenségnek egy {*„A barátom elment.", „A barátom, akihez a szomszédja be szokott csöngetni, elment.", „A barátom, akihez a szomszédja, akinek kölcsönadtam egy százast, be szokott csöngetni, elment.", „..."}* megnyilatkozáshalmaz felelne meg, Igen ám, de már a harmadik mondat nem igazán érthető, és az is könnyen belátható, hogy tetszőleges méretű szövegkorpuszban keresve sem sok az esély, hogy találjunk ilyen szerkezetet. Már pedig csak ennek a szerkezetnek a kedvéért bevezetni a korlátlan mélységet biztosító végtelent meglehetősen nagy „luxus", miközben az $n=1$ és az $n=2$ eseteken kívül ezt a jelenséget a természetes nyelvek nem használják. Chomsky itt az általa bevezetett híres kompetencia–performancia megkülönböztetésre hivatkozik, mondván, hogy az emberi információfeldolgozás fiziológiai–pszichikai esetlegességei nem tartoznak a formálisan jól jellemezhető kompetencia körébe. A kérdés már csak az, hogy a nyelvi kompetencia valóban az-e, amit a formális nyelvészet eszközeivel elegánsan tudunk modellálni?

Ha tehát a reguláris nyelvtanok nem elegendőek a természetes nyelvek leírásához, bizonyára a környezetfüggetlenek elegendőek lesznek hozzá, gondolhatnánk – ám ennek az elképzelésnek is megszülettek a természetes nyelvi példákkal való cáfolatai [4][5]. Az egyik kedvelt érvelés a holland és a svájci-német nyelvre hivatkozik, amelyekben jól kimutathatók bizonyos nem-projektív (azaz: nem szabályosan, egymásba ágyazottan zárójelezhető) szerkezetek. Ezt egy a legtöbb természetes nyelvben, így a magyarban is megtalálható jelenség, az ún. „rendre"-szerkezet egy példáján mutatjuk be: *A MoBiDic, a MoBiMouse és a MoBiCAT programok rendre 1993-ban, 1998-ban és 2003-ban készültek el.* Ennek a mondatnak az az érdekessége, hogy a három név–időpont argumentumpár csak nem-projektív fával írható le. Márpedig a környezetfüggetlen világban ennek a leírása nem lehetséges, amiből egyenesen következik, hogy ha van ilyen természetes nyelv, akkor a természetes nyelvek általánosságban nem lehetnek környezetfüggetlenek. Kérdés ezek után, hogy a természetes nyelvek környezetfüggőek-e, és ha igen, „mennyire"? A modern nyelvészet sokféle megoldást kitalált a környezetfüggetlen nyelvtanok még jól kezelhető bonyolultságának fenntartására. Ravasz megoldások, sokszor ügyes, intellektuális trükkök jöttek létre, melyek a környezetfüggetlen nyelvtanok előnyös tulajdonságait voltak hivatva megtartani, immáron már a környezetfüggő nyelvek világában. Ezeket a rendszereket „slightly context-sensitive"-nek, azaz „éppenhogy környezetfüggőnek" is nevezték.

Egy másik paradigma, a gyakorlati problémákkal gyakran szembesülő számítógépes nyelvészet a generatív nyelvészet kialakulása után hamar létrehozta a maga modelljeit. Ezek az elméleti nyelvészet modelljeitől elsősorban abban különböztek, hogy nem a megnyilatkozások előállítására, hanem a felismerésére koncentráltak. Itt a környezetfüggő modell átugrásával, a reguláris nyelveket leíró véges automaták általánosításán (a rekurzív átmenethálón, az RTN-en) keresztül egyenesen a Turing-gépekkel tették ekvivalenssé mo-

delljeiket, a bővített átmenethálókat, az ATN-eket [6], melyben az állapotátmenetek feltételei közé néhány – a nyelvfeldolgozó feladat megoldásához elengedhetetlennek tűnő – technikai műveletet is felvettek. A természetes nyelvek bonyolultságával kapcsolatos ismereteinkre e gépi modellek megjelenése nem volt, nem lehetett hatással. Ha azonban visszatérünk az elméleti nyelvészet bonyolultsági problémáihoz, könnyen megérthetjük, hogy a Chomsky-modellek tanulmányozása kapcsán hamar megjelent két – egy nyelvészeti és egy matematikai jellegű – dilemma. A nyelvészeti probléma az volt, hogy sok általános nyelvészeti elv megragadására a Chomsky-hierarchia alapnyelvtanai nem alkalmasak. A természetes nyelvek objektumai (azaz például az ugyanazon szavakból álló kijelentő és kérdő mondatok) között „rokonságok" vannak. Chomsky első igazán jelentős – és már idézett – nyelvészeti munkájában megérezte azt az igényt, hogy ezeket a rokonsági relációkat ki kell valahogy fejezni. Bevezette tehát a transzformációt – és így a végtelen számú szabály lehetőségét – először transzformáció-családok formájában, melyből idővel csak egyetlen – ám gazdagon paraméterezhető – elem maradt: a híres „move α" szabály). Peters és Ritchie kimutatta, hogy a transzformációs nyelvtanok gyengén ekvivalensek a Turing-géppel. [7] Később megjelentek a további általánosításokat lehetővé tevő X-vonás (részletesen ld. [8]) és az ID/LP [9]) nyelvtanok. Megjelentek a strukturált kategóriák, az ezekhez szükséges unifikációs műveletek és az őket kezelő szabályosztályok. Ez utóbbiak segítségével – a nagy bonyolultságot magukban hordozó transzformációk nélkül is – kezelhetővé vált több, mindig is nagyon kritikusnak számító jelenség: a távoli függőségek (azaz: a mondaton belül szétesők szerkezetek) és a nem-projektív konstrukciók. A fent jelzett matematikai probléma ugyanakkor Chomsky modelljeiben az volt, hogy a nyelvtanok generálta mondathalmazok összehasonlításán alapuló gyenge generatív kapacitás nem tudta megragadni az „igazi" bonyolultságot.

Egy másik nyelvleírási felfogás, a kétszintes morfológia [10], mely a nyelvnek szóalaktanát és nem elsősorban mondattanát célozza meg, olyan eszközt ígér a nyelvésznek, mellyel az környezetfüggő nyelvtani jelenségeket írhat le, miközben az így készült leírást elemzésre és generálásra használó eszköz formalizmusa megmarad a reguláris nyelvek szintjén. Ebben a rendszerben a lexikális és a felszíni szerkezetek között nincs köztes szint, és a szabályalkalmazás mikéntje – melyet kizárólag a formalizmus működtetésére szolgáló gépi háttérnek és nem a nyelvésznek kell ismerni – garantálja a hatékony működést. A gyakorlat szintjén ez így is látszott lenni, ám egyszerűnek látszó nyelvi jelenségek kétszintes leírása meglepően bonyolult tud lenni. Barton, Berwick és Ristad [11], valamint mások is kimutatták, a bonyolultság itt sem a végső formalizmusban, hanem az azt készítő rendszerben van „elrejtve".

3 Halmaz-e a jó nyelvmodell alapja?

Ha feltesszük, hogy a természetes nyelv halmaz, akkor valóban modellálható rekurzív felsorolással? Mivel tudjuk, hogy létezik nem rekurzívan megszámlálható véges halmaz, csak egy olyan nyelvet kell keresnünk, melyre egy ilyen definíció ráillik. Vegyük például az egyetlen mondatból álló $L=\{z: String(z) \wedge (\forall V)(Over(z,V) \rightarrow V=x) \wedge Length(z)= \aleph_0\}$ nyelvet. Mivel ez a nyelv csak ilyen nyelvtannal írható csak le, ennek az a következménye, hogy ha a „Minden mondat véges objektum" Chomsky-axióma a fenti $\aleph_0$ hossz miatt nem áll, akkor következésképp azzal a Chomsky-axiómával is probléma lesz, hogy „Ez az összesség halmaz". Ennek jobb megértéséhez eljátszunk azzal a gondolattal, hogy valójában hány mondat lehet is egy természetes nyelvben. vegyük ehhez a magyar nyelvet (amely zárt az alá- és a mellérendelésre), és jelöljük L-lel. Ekkor az $S_0=\{$*„Józsi boldog", „tudom, hogy Józsi boldog", „tudom, hogy tudom, hogy Józsi boldog", ...*$\}$ halmaz segít-

ségével létrehozzuk az S_1 halmazt az alábbiak szerint. Ha $P(S_0)$ jelöli az S_0 hatványhalmazát, akkor minden $P(S_0)$-beli B-re legyen S_1 a B összes mondatából álló mellérendelő összetétel halmaza: S_1={*„Józsi boldog", „tudom, hogy Józsi boldog", „tudom, hogy tudom, hogy Józsi boldog", ...; „Józsi boldog és tudom, hogy Józsi boldog", „Józsi boldog és tudom, hogy tudom, hogy Józsi boldog, ...; „Józsi boldog, tudom, hogy Józsi boldog és tudom, hogy tudom, hogy Józsi boldog",...*}. Ekkor tehát – ahogy Cantor tételéből tudjuk – S_0 megszámlálható, de S_1 nem. Ugyanakkor S_2, S_3 stb. ugyanígy létrehozható, egyre növekvő számossággal, viszont minden ilyen S_i eleme L-nek, tehát: az L (magyar) nyelv mondatai nem rekurzívan felsorolhatók, azaz a természetes nyelvek nem írhatók le halmazokként. Ez az állítás pedig valóban ellentmond „A természetes nyelvi mondatok összessége halmaz" Chomsky-axiómának.

Egy másik megfigyelés a mondat hosszával kapcsolatosan gondolkodtathat el. A „Minden mondat kevesebb, mint k elemből áll *($k \in N$)*" állításról intuitíve érezhető, hogy nem igaz, de hogy állunk a „Minden mondat kevesebb, mint $\aleph_0$ elemből áll" állítással? Könnyen látható, hogy a fenti S_1 halmaz mondatainak konjunkciójából előálló mondatra (2) nem igaz! Tehát – amint ezt Langendoen és Postal [12] megmutatja – nem minden mondat véges, azaz ez a Chomsky-axióma nem tartható. Itt ismét jogosan tehető fel a kérdés: ez a modell valóban nehézségeket mutat, de fontos-e egy olyan modell mellett kitartani a végsőkig, mely magáról a tényleges természetes nyelvekről nem mond semmi „negatívat", mindössze a formális megfogalmazás miatt kerül a definíció készítője nehéz helyzetbe [13]. Intuitíve könnyen belátható, hogy az emberi nyelvek leírásához nincs szükség a – mondjuk – egy emberéletnél hosszabb mondatok kimondhatóságáról elgondolkoznunk.

4 A természetes nyelvek bonyolultak, vagy csak a nyelvtanaik?

A bonyolultságelmélet egy adott természetes nyelvi modell esetében megmondhatja például, hogy mennyi ideig tarthat egy nyelvtani probléma feldolgozása. Ehhez nem feltétlenül csak négy grammatikatípust feltételez, mint a Chomsky-hierarchia. Ugyanakkor a bonyolultságelmélet meg tudja mondani, ha például a véges állapotú automata használata nem garantálja a hatékony feldolgozhatóságot, vagy segít a párhuzamos alkalmazhatóság kérdését is körüljárni – *párhuzamos gép megvásárlása nélkül*... Egy formalizmusról végül is nem csak azt szeretnénk megtudni, hogy mennyire bonyolult, hanem a bonyolultságelmélet segítségével esetleg azt is, hogy miért.

Az univerzális nyelvfelismerési probléma, melynek bonyolultságát szeretnénk most meghatározni, így hangzik: adott egy G nyelvtan (valamely nyelvtani formalizmusban meghatározva) és egy a füzér, mi pedig azt szeretnénk megtudni, hogy ez a füzér benne van-e a G által generált nyelvben. A megoldáshoz segítségül hívunk egy ismert bonyolultságú problémát, az ún. 3SAT problémát. Ez azt kérdezi, hogy egy tetszőleges Boole-formula betűihez létezik-e olyan igaz/hamis hozzárendelés, amire az egész kifejezés igaz. Vegyünk egy tetszőleges Boole-formulát, mely három propozícionális változót tartalmaz: $(a \vee \neg b \vee \neg c) \wedge (a \vee b \vee c) \wedge (a \vee b \vee \neg c) \wedge (\neg a \vee b \vee c)$. A megoldáshoz végig kell próbálnunk az összes lehetséges igazságérték-hozzárendelést és megnéznünk, kielégítik-e a formulát, azaz igaz-e az adott állításokkal az egész formula? Ez más szavakkal azt is jelentheti, hogy legrosszabb esetben az összes lehetséges hozzárendelést végig kell próbálnunk, ami n darab bináris változó esetén éppen 2^n lehetséges igazságérték-hozzárendelés. Mivel a változók száma a formula hosszával arányos, a 3SAT probléma megoldása exponenciális időt igényel. Hogy közelebb kerüljünk eredeti célunk, az általános nyelvfelismerési probléma bonyolultságának megvizsgálásához, ezt a formulát először átkonvertáljuk egy a természetes nyelvi szerkezetekhez hasonlóbb formára. Legyen az a az *apple*, a $\neg a$ az

apples, az *b* a *banana*, a $\neg b$ a *bananas*, a *c* a *carrot*, a $\neg c$ a *carrots*, a *d* a *dandelion*, a $\neg d$ a *dandelions*, szó, azaz: „*apple bananas carrots, banana carrot dandelion, apple carrot dandelions AND apples banana dandelion*". Minderről azt állítjuk, hogy ez akkor és csak akkor grammatikus, ha minden részmondatban van „ige", azaz egy *s*-re végződő szó. Mivel a 3SAT probléma is NP-teljes, akkor ez a nyelvtani kérdés egy NP-teljes nyelvtani problémát fed. Az ismét egy másik kérdés, hogy felmerül-e ez a kérdés egyáltalán a természetes nyelvek feldolgozásakor, és másik modellel esetleg az egész probléma megoldható.

A különböző nyelvleíró modellek bonyolultsága természetesen más és más. A hagyományos környezetfüggetlen grammatikák polinomiális időben feldolgozhatóak, ám mind a modern nyelvelméletek, mind a humán nyelvtechnológiák komplexebb, elsősorban a szófaji kategóriák belső szerkezetét jobban leíró modellekkel dolgoznak. Ezek a formális nyelvészetben népszerűbb indexnyelvtanok [14], illetve a környezetfüggetlen vázú unifikációs nyelvtanok [15] és a modern nyelvészetben széles körben használt lexikális-funkcionális nyelvtanok [16] az NP-teljes kategóriába sorolódnak. Sőt, ebbe az osztályba tartozik a véges állapotú automatákat használó kétszintes morfológia is. Ez utóbbi null-elemeket is kezelő verziói már a PSPACE osztályba sorolódnak a hagyományos környezetfüggő grammatikákkal együtt. A transzformációs felismerés még a CS-felismerésnél is bonyolultabb (EXP), de a transzformációs nyelvtanok kiváltására létrejött ID/LP formalizmus, ahol a függőségi hierarchia és a szabálybeli elemek lineáris rendje különválasztva szerepel – az EXP-POLY bonyolultsági osztályban található.

5 Van-e más út?

A modern, formális nyelvelméleti leírások legtöbbje – mint láttuk – olyan matematikai bonyolultságokat hordoz, amelyek, ha előjönnének a valódi, emberi nyelvhasználat során, komoly problémákat jelenthetnének. Ám a nyelvészeti kutatások ilyesmiről soha nem számoltak be, így joggal állapíthatjuk meg, a nyelvleírásra szolgáló egyes formalizmusok bonyolultságáról van szó, nem pedig az emberi nyelvhasználatéról. Sőt, az is igaz, hogy az emberi nyelvfeldolgozás olyan problémákkal is találkozik, melyek megoldására ezek a formális modellek nem is kíséreltek meg megoldást adni. Összefoglalva azt mondhatjuk, hogy vannak bizonyos szempontból jól működő, máshol hiányos modelljeink az emberi nyelvek leírására. Ezeknek a modelleknek a jelentős része matematikailag bonyolult, míg egyes, a számítógépes nyelvfeldolgozás szempontjából lényeges kívánalmaknak nem tesz eleget. A nyelvtechnológiai alkalmazások például – az emberi nyelvfeldolgozási képességek analógiájára – igénylik a bővíthető, tehát az új tulajdonnevek, idegen szavak kezelését lehetővé tevő, azaz: nem zárt lexikon használatát. Ha viszont a lexikon véges, de nem zárt összesség (azaz: létezhetnek nyílt osztályok is a lexikonban), akkor ez ellentmond Chomsky 1957-ben lefektetett negyedik axiómájának. A felsorolható, tehát hagyományosan kezelhető osztályok, azaz a zárt kategóriák az így létrejövő minimálnyelvtan [17] egyik fontos elemét alkotnák, hiszen ezek nélkül a kategóriák nélkül (pl. segédigék, módosítószók, hangsúlyminták) nincs nyelvismeret. A minimálnyelvtan maga nem hoz létre elemeket, hanem leírja azokat – tehát nem-konstruktív nyelvtan. Különösképpen nincs szerepe semmilyen, különösképp nem egyértelmű dichotóm döntés a nyelvi elemekből képezhető objektumok bármilyen mondat-, illetve „nem-mondat"-osztályba tartozásáról.

A nyelvészeti konstrukcióknak az utóbbi évtizedben kikristályosodott fogalma a valamiképpen már előre elkészített nyelvi szerkezeteket kívánja a strukturalizmus nyelvészeti hagyományaira építeni [18]. A nyelvészeti konstrukciók akkor modellálhatók véges leírásokkal, ha a nyelv használatában oly sokszor jelentkező elv, az analógia kezelésére létezik valamiféle mechanizmus. Ez azt jelenti, hogy véges lexikonok fölött véges sok véges

hosszú objektum leírásához elég az ügyes szótárszerű tárolás, a rekurzióra az elemzési módszerekhez van szükség általában. Ha a rekurziót „száműzzük", valamilyen eszközre szükségünk van, ami a véges listában nem szereplő – ritka, de teljes biztonsággal soha ki nem zárható – elemek kezelését biztosítja. A hatalmas véges listával modellálható nyelv elemzési bonyolultsága viszont teljesen más problémákat kell, hogy felvessen, mint amilyeneket a jól ismert formális nyelvek felismerési bonyolultságának kezelésénél láttunk. A hagyományos lexikális kategóriák által hordozott ismeretek konfliktushelyzetben vagy nem használhatók, vagy redundánsak – harmadik eset nincs. Így egy efféle, egyelőre legfeljebb kialakulófélben levőnek nevezhető rendszer [19] bonyolultságának meghatározásához talán az emberi nyelvfeldolgozásra jellemzőbb mérőmódszert lehet a közeljövőben találni, mint a természetes nyelvek leírásához manapság használt formális nyelvek matematikai bonyolultságáét.

6 Hivatkozások

1. Prószéky G. *Számítógépes nyelvészet (Természetes nyelvek használata számítógépes rendszerekben)*. Számalk, Budapest (1989)
2. van de Koot, H. The Computational Complexity of Natural Language Recognition: A Tutorial Overview. *Lingua* (6) (1995) 49–83.
3. Chomsky, N. *Syntactic Structures*. Mouton, The Hague. [Magyarul: Chomsky, N.: *Mondattani szerkezetek – Nyelv és elme*. Osiris–Századvég, Budapest, 1995] (1957)
4. Pullum, G. On Two Recent Attempts to Show that English is Not a CFL. *Computational Linguistics* 10/3–4 (1984) 182–186
5. Shieber, S. (1985) Evidence against Context-Freeness of Natural Language. *Linguistics & Philosophy* 8(3), 333–343
6. Woods, W.A. Transition Network Grammars for Natural Language Analysis. *CACM 13(10)* (1970) 591-606
7. Peters, S., R. Ritchie. On the Generative Power of the Transformational Grammars. *Information Science* (6) (1973) 49–83.
8. Kornai, A. Natural Languages and the Chomsky Hierarchy. *Proceedings of the 2nd Conference of the European Chapter of the ACL*, Geneve (1985), 1–6.
9. Pullum, G. Word Order Universals and Grammatical Relations. In: P. Cole & J. Sadock (szerk.) *Syntax and Semantics* Vol. 8. Academic Press, New York (1976) 249–277
10. Koskenniemi, K. *Two-level Morphology: A General Computational Model for Word-Form Recognition and Production*. Publications No. 11, University of Helsinki, Helsinki (1983)
11. Barton, E.G., R. C. Berwick, E.S. Ristad. *Computational Complexity and Natural Language*. MIT Press, Cambridge, Mass. (1987)
12. Langendoen, D., P.M.Postal. *The Vastness of Natural Languages*. Blackwell, London (1984)
13. Prószéky, G. Review on Langendoen and Postal's "The Vastness of Natural Languages". *Studies in Language* 10(2), 520–527 (1986)
14. Koster, C.H. Affix Grammars. In: Peck, J.E.L. (szerk.) *Algol 68 Implementation*. North-Holland, Amsterdam (1971)
15. Shieber, S. Using Restriction to Extend Parsing Algorithms for Context-Free-Based Formalisms. *Proceedings of the 23rd Meeting of the ACL* (1985) 145–152
16. Kaplan, R. & J. Bresnan. LFG: A Formal System for Grammatical Representation. In: J. Bresnan (szerk.) *The Mental Representation of Grammatical Relations* (1980) 173–281.
17. Kálmán L. & Prószéky G. FMR Grammar. *Working Papers*. Vol.1., Institute of Linguistics, Budapest, 31–41 (1985)
18. Kálmán L. *Konstrukciós nyelvtan*. Tinta, Budapest (2001)
19. Prószéky, G. Morphological Analyzer as Syntactic Parser. *Proceedings of the 16th International Conference on Computational Linguistics*. Copenhagen, 1123–1126 (1996)

On Formal Models of Natural Languages

Gábor Prószéky

MorphoLogic
Budapest
proszeky@morphologic.hu

The paper's aim is to give an overview on how formal models are applied to human language description. Complexity of formal models does not necessarily show the real complexity of natural languages. Language model and language are frequently mixed in the literature. Hence, problems of the well-known formal treatment of language complexity can lead to misleading conclusions.

Új korpuszstatisztikai eszköztár kollokációkeresésre

Kis Balázs, Ugray Gábor

MorphoLogic
{kis,ugray}@morphologic.hu

Kivonat

A nyelvi erőforrások – korpuszok, lexikonok – előkészítése a számítógépes nyelvészet leginkább munkaigényes művelete, szakmai konferenciákon mégis viszonylag kevés előadás foglalkozik vele – talán mert tudományos szempontból itt lehet felmutatni a legkevesebb új eredményt. Ez az előadás is leginkább azt hangsúlyozza, hogyan lehet meglevő nyelvtechnológiai és egyéb számítógépes eszközök felhasználásával hatékonyabbá tenni a korpuszok előkészítését és feldolgozását. Az előadás olyan új korpuszelőkészítő és -statisztikai eszköztárt mutat be, amely általánosan használható kollokációkeresésre egynyelvű korpuszokban, és annotálatlan korpuszból kiindulva is ad értékes eredményt. Az eszköztár a teljes műveletsort felöleli, a korpusz előkészítésétől a statisztikai számítások kiértékeléséig.

1. A feladat

A kollokációkeresés során a leginkább munkaigényes feladat a nyelvészeti erőforrások – korpuszok – előkészítése és a kollokációjelöltek kivonása a szövegből. Ennek sokszor nem tulajdonítunk tudományos jelentőséget, mert feltételezzük, hogy kizárólag mechanikus munkáról van szó – ez azonban távolról sem igaz az általános esetben, amikor a kollokációkutatáshoz nem rendelkezünk kellőképpen előkészített korpusszal.

A kollokációkutatás alapvető fontosságú eleme a számítógépes nyelvészeti munkának, mert a legfőbb alkalmazások – a tartalomelemzés és a fordítástámogatás, illetve az ezek alapjául szolgáló szintaktikai elemzés – megkövetelik a többszavas lexémák, az idiómák, az igevonzatok és más rögzült frazémák megfelelő felismerését.

Ha a kész, nyelvileg alaposan annotált korpuszt mindenféle kollokációstatisztikai művelet előfeltételének tekintjük, akkor jelentős mértékben korlátozzuk a kutatásunkhoz felhasználható korpuszok körét. Ha azonban rendelkezünk olyan eszköztárral, amely hiányosan vagy egyáltalán nem annotált korpuszból is képes előállítani a kollokációjelöltek halmazát, már biztosak lehetünk abban, hogy rövid idő alatt előteremthetjük a statisztikailag is értékelhető nagyságú szövegbázist. (Előadásunkban nem szólunk a korpuszgyűjtés sajátos kérdéseiről.)

A fentiekből látható, hogy a szerzők véleménye szerint értékes kollokációkutatás csak az úgynevezett típusos kollokációk, illetve kollokációjelöltek feldolgozásával lehetséges. A típusos kollokáció mint lokális terminus technicus azt jelenti, hogy a

kollokációk (n-gráfok) komponensei szintaktikailag, morfoszintaktikailag determinálva vannak. Ez az angol vagy a holland nyelvben például az igék és az elöljárószói csoportok (*V + PP*) együttállásának vizsgálatát jelenti, míg ennek – legalábbis egy jelenleg is folyó nemzetközi projekt előfeltételezése szerint – a magyarban az igék és az esetragos, illetve névutós főnévi csoportok együttállása felel meg. (Villada-Bouma 2002) A típusos kollokációk hangsúlyozása azért fontos, mert számos esetben, így például terminológiakivonatoló rendszerekben megelégszenek a típus nélküli kollokációk gyűjtésével, vagyis kizárólag a felszínen előforduló lexémák együttállását vizsgálják, azok morfoszintaktikai jellemzőinek figyelmen kívül hagyásával. (Castellví et al. 2001)

2. A korpusz

Eszköztárunk részben kényszer hatására jött létre. Amikor egy statisztikai csomagot (lásd később!) saját munkakörnyezetünkhöz kellett adaptálnunk, szükségünk volt olyan magyar adatsorokra, amelyek segítségével ki lehetett próbálni. Az egyetlen korpusz, amely e munka közben a rendelkezésünkre állt, a nemrég befejezett SZAK-korpusz volt, amely akkori állapotában nem rendelkezett nyelvi annotációval (Kis-Kis 2003).

A SZAK-korpusz informatikai szakszövegeket tartalmazó párhuzamos korpusz, amely komponensenként kb. 1,2 millió szövegszóra rúg. Ennél lényegesen nagyobb magyar nyelvi korpuszok is rendelkezésre állnak, azonban ez a terjedelem szaknyelvi korpusz esetén elfogadható, eléri a statisztikai számításokhoz szükséges kritikus tömeget.

Az előadásban ismertetett eszköztár a SZAK-korpusz magyar nyelvi komponensét használja fel, azonban az eszköztár első néhány tagját a teljes korpusz formátumának egységesítésére és nyelvi annotálására is felhasználtuk.

3. Az eszköztár

Az eszköztár a korpusz előkészítésének minden műveletét felöleli, a formátum egységesítésétől a végső adatsor összeállításáig. Az alapvető cél az NSP statisztikai csomag (Pedersen-Banerjee 2003) kiszolgálása volt, így az adatsor a statisztikai csomag által megkövetelt formátumot állítja elő.

Az eszköztár által megvalósított protokol főbb elemei a következők:

1. A korpuszbeli szövegek formátumának egységesítése;
2. A szövegek részleges nyelvi elemzése, a típusos kollokációjelöltek kivonatolása;
3. A kollokációjelöltek által reprezentált események heurisztikus utószűrése
4. Az eseménytípusok megszámlálása
5. A statisztikai csomag által megkívánt formátum előállítása

3.1. A korpusz előkészítése

A korpusz előkészítésének legfontosabb lépése a konzisztens – s ilyenformán jól feldolgozható – formátum előállítása. Erre az eszköztár olyan XML-struktúrát alkalmaz, amelyben egyaránt ábrázolhatók a nyelvi annotációval ellátott, illetve az annotációval nem rendelkező szövegek. Ezt a formátumot az eszköztár robusztus módon, közismert állományformátumokból is automatikusan és hibátlanul elő tudja állítani.

Olyan XML-formátumot választottunk, amely az eredeti dokumentum formázásából csak a legfontosabb elemeket őrzi meg, így például azonosítja a címsorokat, és megtartja a címfokozatokkal kapcsolatos információkat. Nem őrzi meg a táblázatokat és képeket, és általában feltételezi, hogy folytonos szövegről van szó. Ezt a formátumot a 2.-ben említett konkrét korpusz előkészítésekor dolgoztuk ki (Kis-Kis 2003).

3.2. Az adatsor kivonatolása

Ha kollokációkeresésről beszélünk, a második fontos lépés a kollokációjelöltek kivonása a korpuszból; ezeket a jelölteket kell később – például statisztikával – értékelni abból a szempontból, hogy valóban kollokációt alkotnak-e.

A kollokációkivonatolás az itt ismertetett eszköztárban a nyelvtechnológia legteljesebb körű kihasználásával történik. A kivonatolás során párokat (bigráfokat), illetve hármasokat (trigráfokat) lehet kiemelni a szövegből; a bigráfok és a trigráfok elemeit morfoszintaktikai, szintaktikai szempontok alapján már a kivonatolási szabályok segítségével szűrni lehet. Ezt az eszköztár úgy éri el, hogy szintaktikai elemző programot alkalmaz (a szerzők HumorESK, illetve Moose nevű eszközét, rendre magyar, illetve angol nyelvű korpuszokhoz), s a kivonatolás során az elemző programok által adott eredményeket (egyes részfák gyökereit) is felhasználja.

A kivonatolás során fontos szempont volt, hogy a későbbi mérési eredmények kompatibilisak legyenek annak a kutatócsoportnak az eredményeivel, ahonnan az NSP-csomagot közvetlenül átvettük. Így a keresés elsősorban a holland *V+PP* kollokációknak leginkább megfelelő magyar *V+NP+case* minták kigyűjtésére, vagyis az igék és az esetragos, illetve névutós főnévi csoportok együttállásának vizsgálatára irányult.

A megfelelő kollokációjelölteket úgynevezett metaszabályok segítségével lehet meghatározni. A mondatelemző programhoz olyan utószűrő modult illesztettünk, amely e metaszabályokat értelmezve a mondatelemző által létrehozott egyes részfák relatív gyökérszimbólumait, illetve a szimbólumok egyes jegyeit szűri ki, és ezeket mint komponenseket használja fel bi-, illetve trigráfok létrehozására. Példa metaszabályra:

```
VX!(lex),NP-FULL!(lex,case):4
```

A fenti szabály olyan trigráfok létrehozását írja elő, amelyek komponensei:

1. egy VX (ige) típusú szimbólum lemmája (*lex* jegye)
2. egy NP-FULL (főnévi csoport) típusú szimbólum lemmája (*lex* jegye) és
3. ugyanazon NP-FULL szimbólum esetragja (*case* jegye)

A trigráfot a rendszer akkor tekinti érvényesnek, ha a VX és az NP-FULL szimbólumok egy 4 terminális pozíciót fedő ablakon belül fordulnak elő együtt. Ez az ablakméret vitatható, mivel így a kivonatolási folyamat figyelmen kívül hagyja az igevonzatban szereplő összetettebb főnévi csoportokat. Példa a kivonatolás eredményére:

küld üzenet ACC 'üzenetet küld'

3.3. A kivonatolt adatsor utószűrése

A kivonatolás természetesen zajosabb, ha a kivonatolást annotálatlan korpuszon végezzük; sokat segít, ha a korpuszt előbb lemmatizáljuk, illetve egyértelműsített szófaji/morfoszintaktikai annotációval (POS-tagging) látjuk el. Az eszköztárhoz az előadás írása idején illesztjük a szófaji annotáló modult.

A jelöltek között megjelenő zajt azonban jelentősen csökkenthetjük egy utószűrő modul segítségével, amely szintén része az eszköztárnak. Ha a kivonatolás során a program egyes jelölteket tévesen ismer fel, s a tévedések egy része szabályokkal leírható, ezek még a statisztikakészítés előtt eltávolíthatók a jelöltek listájából (az eseménylistából).

A heurisztikus utószűrő is metaszabályokat alkalmaz, méghozzá ugyanazon morfológiai elemző felhasználásával. Ez azt jelenti, hogy kiszűrünk olyan többértelműségeket, ahol a kivonatoló választott egy adott trigráfot, amelynek egy komponense másképp is értelmezhető, s nyelvtudásunk alapján épp a másik értelmezés a gyakoribb. E metaszabályok alkalmazása vitatható, hiszen ez a legközvetlenebb beavatkozás a mérési eredményekbe; megfelelő egyértelműsítő (*POS-tagger*) modul alkalmazása esetén szükségtelenné válhat. Addig is alapszabálynak tekintjük, hogy a metaszabályokkal csak a nyilvánvaló zajt szabad szűrni.

3.4. Az események megszámlálása és a statisztikai csomag által megkívánt adatformátum előállítása

Az NSP statisztikai csomag már rendelkezésre álló gyakorisági adatokra alkalmaz különböző statisztikai függvényeket. Ezért a korpusz előkészítéséhez az egyes események gyakoriságának megszámlálása is hozzátartozik. Eseménynek egy kivonatolt n-gráfot (bigráfot, trigráfot) tekintünk.

Az eszköztárnak ezért része egy robusztus gyakoriságszámláló program is, amelynek legfőbb előnye a méretezhetősége: milliós, milliárdos eseményhalmazból is rövid idő alatt előállítja a gyakorisági listát. Vizsgálataink során a fent említetthez hasonló kivonatolást végeztünk a British National Corpus (BNC) anyagának egyharmadán, körülbelül százmillió szövegszón: a kapott eseményhalmaz gyakorisági listájának előállítása egy átlagosnak tekinthető PC-n, hibakereső üzemmódban kb. 40 másodpercig tartott. Sem a korpuszt, sem az eseményhalmazt nem osztottuk részekre, az átalakított, nyelvileg annotált rész-BNC terjedelme 4 gigabájtra rúgott!

Az NSP-csomagban implementált statisztikai függvények paraméterként nemcsak a teljes n-gráfok gyakoriságszámait követelik meg: az n-gráfok komponenseit is meg kell számolni. Így trigráfok esetén a teljes trigráf gyakorisága mellett fel kell tüntetni az 1. és 2., az 1. és 3., illetve a 2. és 3. komponens együttes előfordulásának gyakori-

ságát is, valamint az egyes komponensek önálló előfordulásainak számát is (amikor nem vizsgáljuk, hogy az illető komponens része-e kollokációnak). Ezeket a fent említett gyakoriságszámláló programmal egyszerűen meg lehetett számlálni, de az eszköztárt ki kellett egészíteni egy olyan programmal is, amely összefésüli a különböző számolási menetek során kapott adatsorokat is. Egy trigráf esetén az NSP-csomag által megkívánt bemenet a következő:

```
hoz<>lét<>SUB<>722 1044 1108 22506
```

Ebben még csak az egyes komponensek gyakorisága szerepel. A kételemű részhalmazok gyakoriságát egy, az NSP-csomag részét képező programmal adhatjuk hozzá az adatsorhoz.

Mind a gyakoriságszámláló, mind az adatokat összefésülő programnak kellően robusztusnak kell lennie, mert az adatsor több millió eseményt is tartalmazhat. Ezekben a programokban a Naszódi Mátyás (MorphoLogic) által kifejlesztett, és Kis Balázs által adaptált GammaTrie nyelviadat-indexelő modult használjuk, amely még a MorphoLogic műhelyében létrehozott programok között is kitűnik gyorsaságával.

3.5. Az események statisztikai értékelése, a jelöltek sorbaállítása

A statisztikai feldolgozáshoz eszköztárunk az NSP statisztikai csomagot (Pedersen 2003) használja, amelynek legfőbb előnye, hogy tetszőleges statisztikai függvény illeszthető hozzá. Az első kísérlet során a jól ismert *log-likelihood*, illetve a Villada (2002) által adaptált és az NSP-hez illesztett *salience*-függvényt használtuk; a holland elöljárószói kollokációk keresésekor (Villada 2002) szerint e két függvény bizonyult a legeredményesebbnek, bár terminológiakivonatolási kísérletekben a kölcsönösinformáció-számítás is jól használható (Jacquemin 2001).

A statisztikai függvények olyan rendezett eseménylistát adnak, ahol a lista élén a legrelevánsabbnak tekintett események, a végén pedig a legkevésbé releváns események találhatók. Ez nem feltétlenül van összhangban az események relatív gyakoriságával; a statisztikai vizsgálatok legnagyobb kihívása, hogy az alacsony gyakoriságú, de releváns eseményeket is észre kell venni, és a statisztikailag értékelt eseménylisták elejére kell rendezni.

Az első kísérleti eredmények kiértékelésekor – az igék és az esetragos főnévi csoportok kollokációinak vizsgálatának esetében – azt láttuk, hogy mindkét felhasznált statisztikai függvény az eseménylista elejére rendezte az olyan 'ál-igevonzatokat', amelyek valójában olyan elváló igekötős igék voltak, ahol az igekötő helyén esetragos főnév jelent meg ('vesz észre'). Ez két alapvető tényt bizonyít:

1. A statisztikai csomag alkalmas a magyar nyelvi környezetben, megfelelően előkészített korpuszokon való használatra.
2. A korpuszelőkészítő eszköztár alkalmas arra, hogy statisztikai feldolgozásra akár annotálatlan korpuszokat is előkészítsen, azonban megfelelő eredményt csak akkor várhatunk, ha a részleges elemzéshez használatos magyar nyelvtant megfelelően átalakítjuk, és a többértelműségeket visszaadó morfológiai elemző program helyett egyértelműsítő szófaji címkéző programot (POS-tagger) alkalmazunk.

4. Összefoglalás

Az előadásban bemutattunk egy új, nyelvtechnológiát is felhasználó korpuszstatisztikai eszköztárat, amelyet összekapcsoltunk egy, a nemzetközi irodalomból ismert és különböző kutatóműhelyekben széles körűen használatos statisztikai csomaggal Az eszköztár kollokációkeresésre (bigráfok, trigráfok statisztikai értékelésére) alkalmas. Konkrét kollokációkeresési példán bizonyítottuk az eszköztár által alkalmazott megközelítés helyességét és a nemzetközi irodalomból átvett statisztikai csomag magyar nyelvi korpuszokra való alkalmazhatóságát.

Az itt leírt kutatás jelentős eredményt hozott a MorphoLogic számítógépes nyelvészeti kutatócsoportja számára is, ugyanis itt eddig nem állt rendelkezésre átfogó statisztikai rendszer, így az alkalmazott nyelvtechnológiai megoldásokat meglehetősen nehéz volt korpuszbeli példákkal igazolni. Azonban az itt ismertetett eszköztár és az NSP statisztikai csomag segítségével a kutatócsoport immár egészséges módon össze tudja kapcsolni a számítógépes nyelvészetben alkalmazott szabályalapú és statisztikai módszereket.

Köszönetnyilvánítás

A szerzők köszönetet mondanak a groningeni egyetem (Rijksuniversiteit Groningen) kutatócsoportjának (Begoña Villada Moirón, Gosse Bouma, Bíró Tamás, John Nerbonne) a statisztikai eszköztár adaptálásában nyújtott segítségükért. Köszönet illeti Naszódi Mátyást a GammaTrie nyelviadat-indexelő modulért, Pál Miklóst, Tihanyi Lászlót és Novák Attilát a Humor morfológiai elemző, illetve HelyesLem lemmatizáló modulért.

Irodalomjegyzék

CASTELLVÍ, M. Teresa Cabré – BAGOT, Rosa Estopà – PALATRESI, Jordi Vivaldi: Automatic Term Detection: A Review of Current Systems. In: *Bourigault, Didier – Jacquemin, Christian – L'Homme, Marie-Claude (eds.): Recent Advances in Computational Terminology.* John Benjamins, Amsterdam-Philadelphia, 2001. pp. 53–88.

JACQUEMIN, Christian (2001): Spotting and Discovering Terms through Natural Language Processing. The MIT Press, Cambridge, MA, USA–London.

KIS Balázs (1997): Mi van a szavakon túl? Nyelvtani szerkezetek felismerése számítógéppel. *Előadás a VII. Országos Alkalmazott Nyelvészeti Konferencián.* Külkereskedelmi Főiskola, Budapest, 1997

KIS, Ádám–KIS, Balázs (2003): A Prescriptive Corpus-based Technical Dictionary. Development of a multi-purpose technical dictionary. In: *Proceedings of COMPLEX 2003*, Budapest.

PEDERSEN–BANERJEE (2003): The Design, Implementation and Use of the Ngram Statistics Package. In: *Proceedings of the Fourth International Conference on Intelligent Text Processing and Computational Linguistics* (Mexico City).

VILLADA, Begona–BOUMA, Gosse (2002): A corpus-based approach to the acquisition of collocational prepositional phrases. In: *Proceedings of EURALEX 2002*, Copenhagen, Denmark.

A Proposed New Tool Chain for Corpus Statistics and Collocation Search

Balázs Kis, Gábor Ugray

MorphoLogic
{kis,ugray}@morphologic.hu

The preparation of linguistic resources, namely, corpora and lexicons, is the most work-intensive activity in computational linguistics. Still, only a few papers are published on this topic at general conferences – perhaps because in this field it is the most difficult to achieve new scientific results. This paper emphasizes how existing linguistic tools can be used to more efficiently prepare and process corpora.

The paper presents a new tool chain for corpus preparation and statistics that is generally suitable for collocation searches in monolingual corpora, and is able to produce valuable results even from unannotated corpora.

This tool chain covers the entire process from corpus preparation to the evaluation of statistic results.

The most important step in corpus preparation is providing a consistent format in order to make the text easy to process. To this end, the tool chain applies an XML format that facilitates the representation of texts either with or without linguistic annotation. The tool chain is capable of derive this format in a robust fashion, even directly from common file formats such as MS Word documents or RTF files.

Considering collocation search, the second important step is the extraction of collocate candidates from the corpus; later on, these candidates must be evaluated – by means of statistics, for example – to see if they form a real collocation.

Within the proposed tool chain, collocation extraction utilizes language technology to the greatest possible extent. During the extraction process, either bigrams or trigrams can be selected from the text; their components can be identified, and the collocations can be filtered based on their morpho-syntactic or syntactic properties. To achieve this, the tool chain applies one of MorphoLogic's two parser systems, either HumorESK or Moose (the former is preferred for Hungarian, the latter for English). Parsing results – roots of subtrees in parse forests – can form components of bigrams or trigrams.

The extraction process produces greater noise if performed on unannotated corpora. Precision can be significantly improved by first lemmatizing and POS-tagging the text. A POS-tagger module is being integrated with the tool chain at the time of writing the paper.

Noise among the candidates can be significantly reduced by applying a post-processor filtering program, also part of the tool chain. If a part of the noise can be described using simple rules, the misclassified candidates can easily be removed from the data set before counting statistics.

Milyen a jó Humor?

Novák Attila

MorphoLogic Kft., Budapest
novak@morphologic.hu

Kivonat. Magyar nyelvű szövegek morfológiai elemzésére elterjedten alkalmazzák a MorphoLogic Kft. által kifejlesztett Humor programot. Bár maga a program hatékony eszköznek bizonyult, a Humor adatbázisának formátumával problémák voltak a karbantarthatóság, az olvashatóság, a javíthatóság és a bővíthetőség szempontjából. Ez az előadás azt mutatja be, hogyan sikerült ezt a problémát az elemzőprogram módosítása nélkül a nyelvi adatbázis többszintűvé tételével orvosolni.
Kulcsszavak: automatikus morfológiai elemzés, nyelvi adatbázis.

A Humor morfológiai elemző

A magyarhoz hasonlóan bonyolult morfológiájú nyelvek számítógépes feldolgozása elképzelhetetlen hatékony morfológiai elemzőprogram nélkül. Magyar nyelvű szövegek morfológiai elemzésére Magyarországon leginkább a MorphoLogic Kft. által kifejlesztett Humor programot alkalmazzák (Prószéky és Kis, 1999). Ennek különböző változatait már több mint egy évtizede használják, és időközben a magyar mellett más nyelvekhez is készültek Humor alapú morfológiai elemzők. Bár maga a program hatékony eszköznek bizonyult, az elemző használhatóságát elsősorban az általa használt morfológiai adatbázis minősége határozza meg. Ez az előadás az elemző rövid ismertetése után egy olyan nyelviadatbázis-leíró rendszert mutat be, melynek segítségével jó minőségű magyar morfológiai adatbázist hoztunk létre a Humor elemzőhöz.

A Humor elemző jellemzői

A program klasszikus 'item-and-arrangement' típusú elemzést hajt végre (Hockett, 1954): egy szóalak lehetséges elemzéseit morfsorozatokként adja meg. A szót felépítő minden morfnak kiírja a felszíni és mögöttes alakját, valamint a kategóriáját (amely strukturált információt is tartalmazhat, de lehet belső szerkezet nélküli címke is). Az utóbbi kettő alapján általában azonosítható, hogy melyik morfémáról van szó. Azoknak a homonim lexémáknak az esetében, ahol a szófaj megadása nem elegendő az egyértelműsítéshez, azt a megoldást választottuk, hogy a lexikai alakhoz egyértelműsítő indexet toldottunk (pl. *szél_légmozgás*/*szél_perem*).

A program belső összetevős szerkezet nélküli lapos morfsorozatokként elemzi a szavakat. Ennek az az oka, hogy a program reguláris szónyelvtant tartalmaz, amely determinisztikus és epszilonmentes véges állapotú automataként van implementálva.

Ez egyrészt jóval gyorsabb, mint egy környezetfüggő nyelvtanon alapuló elemző, másrészt ezzel a megoldással elkerüljük sok irreleváns szerkezeti többértelműség előállítását, amit a megfelelő környezetfüggő elemző generálna (pl. a többszörösen képzett összetett szavak esetében).

Az elemző működése

Az elemző mélységi keresést végez a beadott szóalakon a lehetséges elemzések után. Olyan morfokat keres a szótárában, amelyeknek a felszíni alakja illeszkedik a megadott szó még elemzetlen részére. A lexikon nemcsak morfokat, hanem morfsorozatokat is tartalmazhat, amelyeket az elemző így egy lépésben ismer fel.

Elemzés közben a program kétféle ellenőrzést hajt végre. Egyrészt lokális kompatibilitás-ellenőrzést végez az egymás mellett álló morfok között: ellenőrzi a morfofonológiai és a lokálisan ellenőrizhető morfotaktikai feltételek teljesülését. Az előbbire példa a magyarban a magánhangzó-harmónia, az utóbbira pedig az a megszorítás, hogy névszói toldalékok csak névszótöveket követhetnek. Másrészt azt is ellenőrzi, hogy az elemzést alkotó morfémák a nyelv lehetséges szókonstrukciói egyikét testesítik-e meg (megfelelnek-e az adott nyelv morfológiai konstrukcióit leíró szónyelvtannak). A magyarban például a tő+képzők+ragok alakú morfémasorozatok jól formáltak, ugyanilyen kategóriájú morfémák más sorrendben azonban nem jók. A szónyelvtan nem szomszédos összetevők közötti megszorítások ellenőrzését is lehetővé teszi: pl. a *leg-* felsőfokjelet egy tőle jobbra álló morfémának (leggyakrabban a *-bb* középfokjelnek) engedélyeznie kell, közöttük azonban számos más morféma is állhat.

A Humor nyelvi adatbázisa

A program hatékony működésének az a feltétele, hogy az elemzés közben végrehajtandó ellenőrzések nagyon egyszerű és gyors műveletek legyenek. Ehhez az kell, hogy az adatbázis rengeteg redundáns információt tartalmazzon explicit formában, hogy ezeket ne elemzés közben kelljen kiszámítani. A legfőbb probléma az volt, hogy a MorphoLogicnak nem voltak eszközei az elemző által használt adatbázist alkotó redundáns adatszerkezetek létrehozására és karbantartására. A szomszédos morfok közötti lokális kompatibilitás-ellenőrzéshez használt adatszerkezeteket, az allomorfok (és nem morfémák) leírását tartalmazó lexikonokat és a szónyelvtant definiáló véges állapotú automata leírását egyszerű szövegszerkesztő segítségével kellett létrehozni és karbantartani.

A gép számára optimalizált leírások az emberek számára lényegében olvashatatlanok, és ezért nagyon nehéz őket konzisztens módon karbantartani, módosítani, a hibákat megtalálni és kijavítani. A Humor például kétféle adatszerkezetet használ a lokális kompatibilitás ellenőrzésére: egyrészt bináris tulajdonságvektorokat, másrészt kompatibilitási mátrixokat. Mindkét adatszerkezet nagyon nehezen olvasható és a mátrixok kézzel való konzisztens módosítása lényegében lehetetlennek bizonyult. Ráadásul ha egy tulajdonságot vagy jelenséget (pl. a magánhangzó-harmóniát) egyszer az egyik adatszerkezettel ábrázoltunk, nagyon nehéz áttérni a másik adatszerkezettel

való ábrázolásra. Ennek az volt a következménye, hogy a leírások a fejlesztők legjobb szándéka ellenére is hibásak és inkonzisztensek maradtak.

Ezt a problémát az elemzőprogram módosítása nélkül, a nyelvi adatbázis többszintűvé tételével sikerült orvosolni. Egy olyan nyelviadatbázis-leíró keretrendszert hoztunk létre, amelyben a nyelvész magas szintű, ember számára olvasható formátumú leírást készíthet a leírandó nyelv morfológiájáról. Ez a leírás morfémák és nem allomorfok leírását tartalmazza, és az egyes morfémáknak csak azok a tulajdonságai szerepelnek benne, amelyek nem megjósolhatóak. Mivel ez a reprezentáció nem tartalmaz redundáns információt, könnyű konzisztens állapotban tartani. A leírásnak ezen a magas szintjén könnyen lehet a lexikont bővíteni és javítani. Ebből a leírásból a nyelvész által definiált szabályok alapján a keretrendszer állítja elő azokat a redundáns adatszerkezeteket, amelyeket az elemző használ.

A szóalaktani adatbázis létrehozása

A nyelviadatbázis-leíró keretrendszert használó nyelvész munkája a következő feladatok elvégzéséből áll:

- A nyelv morfémakategória-készletének leírása (szófajok, toldalékkategóriák).
- A tő- és toldalékalternációk megadása: le kell írni azt a műveletet, amellyel az egyes fonológiai allomorfiaosztályokba tartozó tövek lexikai alakjából az egyes allomorfok előállnak. Ennek leírására a keretrendszerben reguláris kifejezéseket lehet használni. Meg kell állapítani, hogy mely morfok váltják ki a váltakozást. Ha a váltakozásnak fonológiai vagy fonotaktikai feltétele van, akkor közvetlenül ezekre a tulajdonságokra lehet hivatkozni. Ha idioszinkratikus lexikai jegyek is szerepet játszanak, akkor ezeket be kell vezetni.
- A morfológiai tulajdonságok feltérképezése: azonosítani kell minden olyan tulajdonságot, amely a nyelv morfológiájának leírásánál szerepet játszik. Ezek különbözőfélék lehetnek: vonatkozhatnak a morféma kategóriájára, egy allomorf hangalakjának, illetve írott alakjának valamilyen morfológiailag releváns jellemzőjére, vagy a morféma által kiváltott idioszinkratikus váltakozásra (pl. tőalternációkra).
- A szomszédos morfok közötti szelekciós megszorítások definiálása: ezeket a megszorításokat egy olyan követelményformula formájában kell leírni, amelyet bármely, a morffal szomszédos más morf tulajdonsághalmazának ki kell elégítenie. A tulajdonsághalmazok és a követelményeket leíró formulák az előző pontban azonosított morfológiai tulajdonságokat tartalmazhatják. Minden morf két tulajdonsághalmazzal rendelkezik: az egyiket a morffal balról, a másikat a morffal jobbról szomszédos morfok látják. Hasonlóképpen minden morf egy-egy formulával megszorítást tehet mind a vele balról mind a vele jobbról szomszédos morfémákra nézve. Egy morfot csak akkor követhet egy másik, ha mind a bal oldali morf jobbról látható tulajdonságegyüttese kielégíti a jobb oldalinak a bal szomszédjával szemben támasztott követelményeit, mind pedig a jobb morf balról látható tulajdonságegyüttese kielégíti a bal oldalinak a jobb szomszédjával szemben támasztott követelményeit.
- A morfémák és allomorfok tulajdonságai közötti implikációs viszonyok megadása: ezeket az implikációs viszonyokat olyan szabályok formájában kell megfogalmazni, amelyek leírják, hogy az allomorfok redundáns tulajdonságai hogyan számít-

hatók ki a már ismert (a lexikonban megadott, vagy korábban már kiszámított) tulajdonságaikból (ide értve az alakjukat is). A szabályok default tulajdonságokat is bevezethetnek mind a morfémák mind az allomorfok szintjén, és a szomszédos morfokra vonatkozó megszorításokat is megfogalmazhatnak. A szabályokat egy erre a célra alkotott viszonylag egyszerű procedurális nyelven lehet leírni. A tő- és toldalékallomorfok előállítását leíró mintákat is a szabályfájlok tartalmazzák.

- A tő- és toldaléklexikonok előállítása: a morfológiai elemző által használt lexikonnal ellentétben a nyelvész által létrehozott lexikonok morfémák és nem allomorfok leírását tartalmazzák. A morfémákat a lexikai alakjuk, a kategóriájuk és a megjósolhatatlan vagy rendhagyó tulajdonságaik és elvárásaik megadásával kell leírni. A rendhagyó toldalékolt alakok és szuppletív allomorfok is megadhatók a lexikonban. Ezek leírásának ez a preferált módja, bár a rendszer azt is lehetővé teszi, hogy nagyon szűk körben működő szabályokkal állítsuk őket elő. A komplex lexikai egységek (elsősorban az összetett szavak) konzisztens és gazdaságos leírásának elősegítésére beépítettünk a rendszerbe egy egyszerű öröklési mechanizmust, amelynek segítségével az összetett lexikai egységek alapesetben az utótagjuktól öröklik a tulajdonságaikat. Az öröklési mechanizmus működésének az a feltétele, hogy a szavakat az összetételi határok jelölésével kell a lexikonba felvenni.
- A szónyelvtan leírása: a szavak belső alaktani szerkezetére vonatkozó megszorításokat (ideértve a nem szomszédos morfémák közötti megszorításokat is) a szónyelvtan írja le. A Humor elemző reguláris szónyelvtan használatát teszi lehetővé. A nyelvtant az elemző számára determinisztikus, epszilonmentes kiterjesztett véges állapotú automata formájában kell leírni. Az automata annyiban kiterjesztett, hogy az egyes állapotátmenetek megadásakor egy véges bináris vagy több bites változókészlet elemeinek értékét lehet módosítani, illetve ellenőrizni. A keretrendszer az automata leírását egyrészt azzal könnyíti meg, hogy szimbolikus változónevek definiálását teszi lehetővé, és ezzel olvashatóbbá teszi a leírást, másrészt egy hatékonyan használható makródefiniáló és –kezelő eszközt is biztosít, amelynek segítségével több hasonló, de részleteiben különböző állapotátmenetet lehet egyszerre definiálni (ami a bonyolultabb automaták leírását nagyban megkönnyíti).
- Külön toldaléknyelvtan létrehozása (nem kötelező): egy irányított gráf formájában külön toldaléknyelvtant lehet definiálni, amelynek felhasználásával a keretrendszer a toldaléklexikonból elemzett toldaléksorozatokat állít elő. Ezeknek az előre megelemzett morfsorozatoknak az elemző lexikonjába való felvétele jelentősen gyorsítja az elemző működését, mert a magyarban és a hozzá hasonló agglutináló nyelvekben nem ritkák a hosszú toldaléksorozatok. A toldaléknyelvtan használatának a másik előnye az, hogy a szónyelvtannak azt a részét, amit a toldaléknyelvtan segítségével leírtunk általában ki lehet hagyni az elemző által használt szónyelvtan-leírásból, aminek eredményeképpen az utóbbi jelentősen egyszerűsödik.

A morfológiai adatbázis átalakítása

A fent leírt módon elkészített leírás alapján a keretrendszer olyan reprezentációt hoz létre, amelyben már minden morféma minden allomorfja az összes tulajdonságával és elvárásával együtt explicit módon szerepel. Az így előálló reprezentáció még mindig olvasható formában tartalmazza az egyes morfok tulajdonságait és szelekciós megszo-

rításait kifejező formulákat, így a nyelvész könnyen ellenőrizheti a leírások helyességét. Az alábbi példa a *kutya* szó redundáns reprezentációját mutatja be.

```
lemma:  'kutya[FN]'
 root:  'kutya'
 allomf:'kutya'
  mcat: 'S_FN'
  rp:   '-Vs -nyi -sÁg -tAlAn =_s =_t =i =jA =vAl VHB
Vfin cat_N cmp2 sfxable mcat_stem'
  rr:   '!FVL'
  lp:   'Cini comp2 k_ini'
  lr:   '!cat_vrb'
 allomf:'kutyá'
  mcat: 'S_FN'
  rp:   '-Vs -nyi -sÁg -tAlAn =_s =_t =i =jA =vAl VHB
Vfin cat_N cmp2 sfxable mcat_stem'
  rr:   'FVL'
  lp:   'Cini comp2 k_ini'
  lr:   '!cat_vrb'
```

A *kutya* tőnek, amely főnév (`[FN]`) kategóriájú két alakja (allomorfja) van: egy *kutya* és egy *kutyá* alakú. A két allomorf jobb, és bal oldali tulajdonságai (`rp` ='right side properties', ill. `lp`='left side properties') valamint a bal oldali elvárásaik (`lr`='left side requirements') is megegyeznek. A jobb oldali tulajdonságok közül a – kezdetűek arra utalnak, hogy a megfelelő képzőt a tő felveheti, az = kezdetű tulajdonságok azt írják le, hogy a megfelelő toldalékot a tő milyen alakban veszi fel. A `Vfin`, `Cini`, `k_ini` a morf alaki tulajdonságait írják le (magánhangzóra végződik, mássalhangzó kezdetű, *k* kezdetű), a `VHB` azt írja le, hogy a harmonikus toldalékok hátul képzett változata kapcsolható hozzá, a `cat_N`, `cmp2`, `sfxable`, `mcat_stem` pedig a morféma kategoriális tulajdonságait írják le (főnév, szerepelhet összetétel második tagjaként, toldalékolható és tő), amelyek – az elemző számára készített redundáns leírásról lévén szó – minden allomorf leírásánál explicit módon megjelennek. A ! a tagadás jele: a `!cat_vrb` megszorítás jelentése: igető után nem állhat. A *kutyá* allomorf jobb oldali szomszédainak `FVL` ('final vowel lengthening') tulajdonsággal kell rendelkezniük, vagyis olyan toldaléknak kell lenniük, amelyik kiváltja a tővégi alsó magánhangzó (*a* vagy *e*) megnyúlását. A *kutya* allomorftól jobbra éppen az ilyen tulajdonsággal bíró morfok nem állhatnak (`!FVL` megszorítás).

Ezt a reprezentációt a keretrendszer a következő lépésben az elemző által használt formájúra alakítja. A fordítás alapjául egy olyan leírás szolgál, amely minden egyes, a nyelv leírásánál használt tulajdonságra megadja a kódolás módját az elemző számára. Lehetőség van arra is, hogy egy tulajdonságot a fordításkor figyelmen kívül hagyjunk, így létre lehet hozni az elemző olyan módosított változatait is, amelyek bizonyos megszorításokat figyelmen kívül hagynak, és ily módon túlelemeznek. A fordítás alapjául szolgáló leírás elkészítése szintén a keretrendszer felhasználójának a feladata.

Az általunk használt egyszerű propozicionális leírás minden tulajdonságot binárisan reprezentál, a leírandó nyelv morfológiája azonban olyan, hogy bizonyos tulajdonságok igaz voltából automatikusan következik, hogy egyes más tulajdonságok nem lehetnek igazak az adott objektumra, ha pl. egy tő ige, akkor nem lehet főnév is egyben. A keretrendszer lehetővé teszi, hogy kifejezzük, hogy bizonyos tulajdonsá-

gok ugyanannak a jegynek egymást kizáró lehetséges értékei. Az ilyen tulajdonságokat valódi független bináris tulajdonságokra dekomponálhatjuk, ami egy konjunktív következményformula (tkp. egy jelentésposztulátum) formájában adható meg a tulajdonság kódolását megadó leírásban.

Az új magyar morfológiai adatbázis

A keretrendszer felhasználásával teljesen új leírást készítettünk a magyar morfológiáról. Az eredeti Humor adatbázisból kizárólag lexikai információt vettünk át: az új elemző tőmorféma-készlete eleinte megegyezett az eredetiével, de rengeteg hibás vagy inkonzisztens kategóriacímkét kijavítottunk, és a komplex (összetett, képzett) tövek szegmentálását megadtuk (erre az öröklési mechanizmus működéséhez is szükség van). A zárt tőosztályokba tartozásra vonatkozó információt (pl. *v*-vel bővülés, tőmagánhangzó-rövidülés, nyitótőség stb.) szintén az eredeti adatbázisból nyertük (javításokkal).

A toldalékok kategóriacímkéi – a kompatibilitás kedvéért – általában megegyeznek a korábbiakkal, de néhány korábban szételemzett toldalékot atominak tekintettünk az új leírásban (pl. a *–hAtÓ* és a *-hAtAtlAn*). A névmás, mint kategória megszűnt: a névszói és határozói kategóriákon belül vannak névmási tulajdonsággal bíró tövek.

Paradigmatikus információt egyáltalán nem vettünk át az eredeti leírásból; a paradigmák az allomorfokat és tulajdonságaikat, illetve elvárásaikat kiszámító szabályrendszer révén állnak elő.

Az eredeti rendszerrel ellentétben az újba nagyon könnyű új szavakat felvenni, mert csak azokat a megjósolhatatlan tulajdonságaikat kell a szótárba felvenni, amelyek különböznek a defaulttól. Ez a szavak túlnyomó többsége esetében a lexikai alakra és a kategóriacímkére korlátozódik, illetve az esetleges összetételi határok megadására (a *kutya* szó reprezentációja a tőadatbázisban például egyszerűen `kutya[FN]`, ebből automatikusan áll elő a fentebb látott redundáns reprezentáció).

A keretrendszer használatával készült egyébként egy jó minőségű spanyol morfológiai elemző is, ezen kívül egy folyamatban lévő projekt keretében számos kisebb finnugor és más uráli nyelv leírására is ezt a rendszert használjuk.

Hivatkozások

C. Hockett. 1954. Two models of grammatical description. *Word* 10 (2): 210–234.

Prószéky Gábor és Kis Balázs. 1999. A Unification-based Approach to Morpho-syntactic Parsing of Agglutinative and Other (Highly) Inflectional Languages. *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, 261–268. College Park, Maryland, USA

What is good Humor like?

Attila Novák

Morphologic Ltd., Budapest
novak@morphologic.hu

Keywords: morphological analysis, linguistic database

Computational processing of highly inflectional languages, like Hungarian, relies upon an efficient morphological analysis. The program most commonly used for the morphological analysis of Hungarian texts is the analyzer called Humor, developed by a Hungarian language technology company, MorphoLogic. Various versions of Humor have been in use for over a decade now. Although the program itself proved to be an efficient tool, the original database format turned out to be problematic, because it was hard to create and maintain. This paper describes how this problem was solved.

The Humor analyzer analyses the input word as a sequence of morphs. It is segmented into parts which have a surface form, a lexical form and a category label. While the program performs a search on the input word form for possible analyses, two kinds of checks are performed at every step: it checks the local compatibility between adjacent morphs and it examines whether the morphemes in the analysis instantiate a possible word construction in the given language.

The operations that the analyzer uses when analyzing the input word must be very simple so that processing can be efficient. This requires that the data structures it uses contain much redundant data (so that they do not have to be calculated on the fly during analysis). The most important problem with the Humor analyzer was that MorphoLogic had no tools for creating and maintaining these redundant data structures. The data structures optimized for efficient manipulation by the analyzer were hardly readable for humans, and they were very hard to modify in a consistent way. This resulted in many errors and inconsistencies in the descriptions, which were very difficult to find and correct.

To solve this problem an environment was created which facilitates the creation of the database. In the new environment, the linguist has to create a high level human readable description which contains no redundant information and which is thus easy to keep consistent. This high level representation makes the maintenance of the lexicon very easy.

This representation is transformed by the system in a consistent way to a redundant, but still readable description using rules defined by the creator of the database. The rules describe allomorphy patterns and implicational relations between morphological properties. At this level of representation, it is easy to catch errors in the rule system. In the next step, the low level representation used by the analyzer is created.

Using the development environment, a completely new version of the Hungarian analyzer was created, which contains less errors and is much easier to maintain than the previous one. A project is under way in which morphological analyzers for various Finno-Ugric and other Uralic languages are created using the system.

Komplex (magyar) szintaktikai elemző rendszer mint beágyazott rendszer

Kis Balázs, Naszódi Mátyás, Prószéky Gábor

MorphoLogic
{kis,naszodim,proszeky}@morphologic.hu

Az előadás az 1995 óta fejlődő HumorESK mondatelemző rendszer fejlődését, új lehetőségeit és alkalmazásait mutatja be. A fejlesztők a HumorESK architektúráját úgy alakították ki, hogy a rendszer a legkülönbözőbb alkalmazások beágyazott komponense lehessen. A program teljesen adatvezérelt, ami azt jelenti, hogy a feldolgozandó nyelvet, a nyelvtant és az elemzés mélységét illetően semmilyen előfeltételezéssel nem él: ennek minden paraméterét a felhasználó határozhatja meg. Az előadás során a szerzők vázlatosan ismertetik azokat az alkalmazásokat és projekteket, amelyekben eddig felhasználták a HumorESK mondatelemző rendszert.

1. A HumorESK program elméleti alapja

A HumorESK program az – ugyancsak a MorphoLogic által kifejlesztett – MetaMorpho (MMO) nyelvtani formalizmus első implementációja. Az előadás 1. része ezért többnyire általában is érvényes a MetaMorpho-formalizmusra, ugyanakkor a rendszer fejlődése során e formalizmus olyan elemekkel is kiegszül, illetve kiegészült, amelyek kimondottan a HumorESK-implementációra jellemzőek.

A HumorESK program nem valósít meg forradalmian új elemzési algoritmust. Alulról felfelé végzi a szöveg elemzését; az egyes szimbólumokhoz egyszerűsített struktúrájú jegyszerkezeteket (*feature structure*) kapcsol, és az elemzési erdő építése során az itt feltüntetett jegyek értékeit ellenőrzi, illetve örökölteti. A HumorESK működésének logikája leginkább a PATR-II formalizmusnak felel meg (Shieber, Uszkoreit, Pereira, Robinson, Tyson 1983).

A HumorESK megvalósításában a szabályillesztés módja és a szabályok megfogalmazása számít újdonságnak. A szabályokat véges mintahalmaz formájában írjuk le, így a rendszer a szabályok illesztéséhez a morfológiai elemzőkéhez hasonló lexikont kap. Az egyes minták alulspecifikált elemekből – szimbólumokból – épülnek fel: egyes szimbólumok esetén csak a szintaktikai szerepet jelző címkét ismerjük, mások esetében meg van adva a lemma vagy éppen a felszíni szóalak is.

A fentiek miatt a HumorESK-et nem lehet egyértelműen a szabályalapú vagy a szótáras – a gépi fordítástól kölcsönzött kifejezéssel: példaalapú – rendszernek nevezni: az elemzési adat, vagyis a nyelvtan elemi komponense egyfelől olyan szabály, amelynek egyes elemeit lexikailag megszorítjuk; ugyanez másfelől olyan minta (példa), amelynek egyes részei nincsenek teljesen, a felszíni jelsorozat szintjén specifi-

kálva. Ez feltételezésünk szerint lehetővé teszi, hogy a rendszer elméletben bármelyik meglevő nyelvtani formalizmussal ekvivalens legyen, különösen ha a rendszerbe bekerül a sorrendfüggetlen szabályillesztés mechanizmusa is.

A nyelvtanban lehetnek ugyanarra a nyelvi jelenségre általánosabb és specifikusabb minták is. Ezért a HumorESK által alkalmazott formalizmusnak fontos eleme a szabályok közötti felülbírálati mechanizmus; a specifikusabb szabály jellemzően felülbírálja az általánosabbat. Ezzel egyfelől csökken a rendszer túlgenerálása, másfelől pedig a kapott elemzésekben az egyes lexikai elemekre (terminális szimbólumokra, „szavakra") vetített elemzések jobban megfelelnek a szimbólumok környezetének. A HumorESK így tulajdonképpen sokszor anélkül is meghatározza az egyes szavak környezetnek megfelelő „jelentését", ha maga jelentés semmilyen formában nincs reprezentálva a nyelvtanban. Példa:

```
NP=ADJ+N:155261
HU.NP[ ... ] = ADJ(...) + N(...)

NE=ANY+NE(nev):16772
HU.NE[...] = ANY( casetype=UPPERINITIAL ) + N( prop = FIRSTNAME )
!155261
```

A fenti esetben mindkét minta illeszkedik a mondatkezdő „Fekete Péter" karaktersorozatra. A második azonban specifikusabb: ez abból látszik, hogy az ott leírt N szimbólumnak rendelkeznie kell a prop tulajdonsággal, annak pedig a FIRSTNAME értékkel. Ez azt jelent(het)i, hogy a jelzett helyen olyan főnévnek kell szerepelnie, amelyet egy korábbi minta vagy éppen a morfológiai elemző modul személynévként azonosított. A második minta kiegészül a `!155261` sorral, ami azt jelenti, hogy amennyiben a minta „elsül", illeszkedik egy bemeneti jelsorozatra, akkor ennek a mintának felül kell bírálnia a 155261 azonosítójú másik mintát, amennyiben az is illeszkedett ugyanarra a bemenetre.

A címkék diverzitása, illetve a jegyszerkezetek kihasználásának mértéke a nyelvtanban tetszés szerint választható meg. A HumorESK architektúrájától nem idegen a mintáknak az elemek felszíni sorrendjétől független illesztése sem – ez a funkció jelenleg fejlesztés alatt áll –, ennek megvalósítása esetén a szimbólumok nemprojektív módon is származtathatók.

Fontos lehetőség a HumorESK-ben, illetve a MetaMorpho-formalizmusban, hogy a szabályokat leképező mintákhoz transzformációk rendelhetők. Így a minták alkalmas megfogalmazása esetén a rendszer már elemzési időben logikai struktúrává „fordíthatja" a szöveget.

2. A HumorESK megvalósításának lényeges vonásai

A fejlesztők a HumorESK architektúráját úgy alakították ki, hogy a rendszer a legkülönbözőbb alkalmazások beágyazott komponense lehessen. A program teljesen adatvezérelt, ami azt jelenti, hogy a feldolgozandó nyelvet, a nyelvtant és az elemzés mélységét illetően semmilyen előfeltételezéssel nem él: ennek minden paraméterét a felhasználó határozhatja meg.

Így lehetőség van teljes mondatok vagy mondatfeletti struktúrák nagy mélységű elemzésére is; ezzel például részletes tartalomelemző alkalmazások működtethetők – a 3. részben ilyen alkalmazást is bemutatunk. Ugyanakkor olyan nyelvtant is készíthetünk, amellyel a HumorESK csak egyes részstruktúrák kis mélységű elemzését végzi el, így alkalmas például NP-kivonatolásra (*NP chunking*), illetve általános kollokációkeresésre is.

Az elemzés mélysége és a bemeneti szegmensek lefedése egyazon nyelvtannal is lehet különböző: a HumorESK-ben a nyelvtan szintekre bontható, a konfigurációban pedig előírható, hogy az elemzés mely szintig történjen meg.

A HumorESK valós idejű rendszer; az elemzési idő korlátozható, s akkor is kiolvashatók hasznos elemzési eredmények, ha az algoritmus logikája szerint még nem fejeződött be a szegmens eredménye.

A HumorESK implementációja statikus programkönyvtárként, C- és C++-illesztőfelülettel áll rendelkezésre. Jelenleg a 32-bites Windows alatti megvalósítás érhető el; a Unix/Linux-rendszerekben használható változat fejlesztés alatt áll.

A mondatelemző program a cikk írása idején az alkalmazásokra jellemző kétféle mélységű magyar nyelvtannal működik. A mélyebb elemzést előíró nyelvtan kb. 20 000 mintát tartalmaz. Ezzel a nyelvtannal egy szegmens (mondatjelölt) elemzése, hibakereső üzemmódban, átlagos PC-n 10-300 ms időt vesz igénybe, az átlagos elemzési idő 50 ms alatt van olyan szövegekben, ahol egy mondatjelölt jellemzően 20 szónál hosszabb.

3. A HumorESK alkalmazásai

3.1. Üzleti rövidhírek tartalomelemzése

2003 közepén zárult le egy NKFP-projekt, amelynek célja üzleti rövidhírek tartalomelemzése volt. Ez olyan alkalmazás – a *NewsPro* – készítését jelentette, amelynek elemeznie kell a rövidhírek mondatait, s a mondatelemzés eredményeire olyan szemantikai kereteket kell illesztenie, amelyek lehetővé teszik az egyes mondatok által leírt események, illetve az események szereplőinek azonosítását. Ha például egy bank megnöveli tulajdonrészét egy cégben, akkor ezt – a tulajdonrész meglétét és növekedését – a rendszernek megfelelően azonosítania kell mint eseményt, és fel kell ismernie, hogy az esemény szereplői között megjelenik a bank mint vevő (és tulajdonos), a cég mint az adásvétel (és a tulajdonlás) tárgya, az eredeti tulajdonrész (ha meg volt adva), és a növekedés mértéke.

Ebben a rendszerben a HumorESK mondatelemző végzi a rövidhírek mondatainak elemzését. Ehhez meglehetősen bonyolult szerkezeteket lefedő, viszonylag nagy mélységű elemzést adó magyar mondatnyelvtant kellett készíteni. Ez a mondatnyelvtan három lényeges komponenst tartalmaz, amelyek külön-külön is jelentős fejlesztést igényeltek, és általában is nagy mértékben járultak hozzá a magyar számítógépes szintaxis fejlődéséhez:

(1) *Tulajdonnév-felismerés.* Az üzleti rövidhírek nagy mennyiségben tartalmaznak személy-, cég-, intézmény-, helyneveket, dátum- és időmeghatározásokat, illetve

pénzösszegeket. Ezek felismerésére kiterjedt résznyelvtan készült, amelyet az MTA Nyelvtudományi Intézetének Korpusznyelvészeti Osztályán készítettek elő a Clark programmal, s a MorphoLogic munkatársai ezt követően adaptáltak a HumorESK számára. Ez a tulajdonnév-felismerő rendszer – annak ellenére, hogy a NewsPro-projekt tesztkorpusza alapján készült – általánosan is használható, s más projektekben, sőt részben más nyelvű nyelvtanokban is megjelenhet tulajdonnév-felismerő modulként.

(2) *Főnévi csoportok felismerése.* Az üzleti hírek szövegeinek megfelelő bonyolult főnévicsoport-nyelvtant kellett készíteni, amely más jellegű szövegekben nem kívánt módon túlgenerálhat, ezért általános szövegekhez még adaptálni kell. Ezzel kapcsolatban viszont jelentős eredmény, hogy a NewsPro-rendszerhez készített nyelvtan a magyar főnévi csoportokban megjelenő legtöbb jelenséget lefedi (beágyazott melléknévi igenévi szerkezetek, különféle birtokos szerkezetek, értelmező jelzők stb.), s mint ilyen, a magyar főnévi csoportok eddigi legteljesebb számítógépes leírása. Elméleti szempontból viszont nem egységes, mivel pragmatikus szempontok szerint, egy igen koncentrált korpusz által reprezentált nyelvváltozat leírására szolgál.

(3) *Igevonzatok felismerése.* A NewsPro-rendszerhez készített HumorESK-nyelvtan az MTA Nyelvtudományi Intézetének Korpusznyelvészeti Osztálya által készített igevonzat-szótár adaptált változatát alkalmazza. Több mint 8000 igevonzatot, illetve egyes esetekben ezek általánosított változatát tartalmazza. Az eddigi magyar számítógépes szintaxisok közül e tekintetben is a legteljesebbnek számít.

3.2. Narratív pszichológiai tartalomelemzés

Egy másik NKFP-projektum keretében a MorphoLogic az MTA Pszichológiai Intézetével közösen pszichológiai narratívumok – interjúk során rögzített, az alanyok által elmondott történetek – elemzését végzi. A nyelvi elemzés feladata itt viszonylag korlátozott: úgynevezett parciális vagy lokális nyelvtanok segítségével meghatározott nyelvi markereket (pszichológiai szempontból jelentős, a narratívumban megjelenő nyelvi jelenségeket) kell felismerni. Itt nincs szükség teljes mondatok mély elemzésére: elegendő meghatározott nyelvi jelenségek jelenlétét észlelni, és a jelenségeket a szövegekben megjelölni. A megjelölt markerek alapján pszichológusok statisztikát készítenek, s ezeket használják fel további kutatási célokra (lásd László et al. 2003, elhangzik ugyanezen a konferencián).

Az elemzés során a HumorESK-nek a következő markertípusokat kell felismernie:

(1) idő és idővel kapcsolatos megnyilvánulások

(2) a közelítés és távolítás kifejezései,

(3) a narratív perspektíva kifejezései.

A megfelelő nyelvtanok kifejlesztése során különösen nagy problémát jelentett az időt kifejező határozók és határozói szerkezetek felismerése; erről ugyanezen a konferencián külön előadás szól (Naszódi 2003).

Ebben a projektben a HumorESK mondatelemző modult egy LinTag nevű programba ágyaztuk, amely az elemzési eredményeket olyan formába alakítja, amely az Atlas.ti nevű statisztikai programcsomagban használható fel.

3.3. Korpuszstatisztikai eszköztár

A HumorESK felhasználásával olyan korpuszstatisztikai eszköztár is készült, amely különösen alkalmas típusos kollokációk keresésére. A korpuszstatisztikai eszköztár oly módon von ki adatokat akár annotálatlan korpuszból is, hogy az az NSP nevű statisztikai programcsomaggal legyen feldolgozható (Pedersen 2003). A korpuszstatisztikai eszköztárt ugyancsak bemutatjuk ezen a konferencián (Kis-Ugray 2003).

A HumorESK modult ezúttal egy parancssori eszközbe (mlc_dataset) építettük, amely a korpuszbeli mondatként szegmentált szakaszok elemzését végzi el, s az eredményekből úgynevezett kivonatolási metaszabályok segítségével kikeresi a releváns származtatott szimbólumokat.

A korpuszstatisztikai eszköztárral végzett kísérleteinkhez a 3.1. alatt említett NewsPro-projekthez készített hét logikai szintből álló magyar mondatnyelvtan alsó három szintjét használtuk fel (az egyszerű főnévi csoportokkal bezárólag). Kihasználtuk a HumorESK azon lehetőségét, hogy az elemzés maximális szintjét futásidőben meg lehet határozni: így, bár a kivonatoló futtatásához eredendően komplex nyelvtant használtunk fel, a felső négy szint kikapcsolása révén a rendszer nem használt fel a szükségesnél több erőforrást – sem processzoridőt, sem tárolóhelyet.

4. Összefoglalás

Ez az előadás a HumorESK mondatelemző modul legújabb alkalmazásait mutatta be, bizonyítva, hogy az eredetileg 1995-ben felvázolt mondatelemzési modell és annak implementációja alkalmas a széles körű felhasználásra.

Köszönetnyilvánítás

A HumorESK alkalmazásainak és a nyelvtanok elkészítéséért köszönet illeti a következő kollégákat: Tihanyi László, MorphoLogic (MetaMorpho-formalizmus), Váradi Tamás és kollégái, MTA NyTI (tulajdonnév-felismerés, névszói csoportok, igevonzatok struktúrájának meghatározása), Benkő Borbála Katalin és Katona Tamás, BME HIT (a NewsPro-mondatnyelvtan szerkesztői és implementálói), Gyimóthy Tibor, Alexin Zoltán és kollégáik, SZTE (a NewsPro szemantikai kereteinek kidolgozása), László János, Ehmann Bea, Pólya Tibor, Pohárnok Melinda, MTA PI (a narratív pszichológiai tartalomelemzés elvi kidolgozása és az elemzési eredmények továbbfeldolgozása), Gosse Bouma és Begoña Villada Moirón, Humanities Computing, Rijksuniversiteit Groningen (az NSP statisztikai programcsomag adaptálása, a korpuszstatisztikai eszköztár kimenetének értékelése).

Irodalomjegyzék

KIS Balázs (1997): Mi van a szavakon túl? Nyelvtani szerkezetek felismerése számítógéppel. *Előadás a VII. Országos Alkalmazott Nyelvészeti Konferencián.* Külkereskedelmi Főiskola, Budapest, 1997

LÁSZLÓ János–EHMANN Bea (2004): Narratív pszichológia és narratív pszichológiai tartalomelemzés (kéziratban). In (várható): Magyar Pszichológiai Szemle, 2004/2., Budapest.

NASZÓDI Mátyás: Nyelvhelyesség-ellenőrzés számítógéppel (parciális szintaxis). Elhangzott a *VII. Országos Alkalmazott Nyelvészeti Konferencián* (Külkereskedelmi Főiskola, Budapest, 1997)

PEDERSEN–BANERJEE (2003): The Design, Implementation and Use of the Ngram Statistics Package. In: *Proceedings of the Fourth International Conference on Intelligent Text Processing and Computational Linguistics* (Mexico City).

PRÓSZÉKY Gábor–KIS Balázs (1999): *Számítógéppel - emberi nyelven.* SZAK Kiadó, Bicske.

PRÓSZÉKY, Gábor (1996): Syntax As Meta-morphology. *Proceedings of COLING-96*, Vol.2, 1123-1126. Copenhagen, Denmark.

PRÓSZÉKY, Gábor (1999): Lexical Information and Decisions in Parsing. In: Cristea, Dan, Dan Tufiş, Amalia Todiraşcu, Valentin Tablan & Cătălina Barbu (eds.) *4th Eurolan Summer School on Human Language Technology, Technical Report 99-02*, ISSN 1224-9327, Iaşi, Romania.

SHIEBER, S. M., H. Uszkoreit, F. C. Pereira, J. Robinson, and M. Tyson (1983). The formalism and implementation of PATR-II. In *J. Bresnan, editor, 23 Research on Interactive Acquisition and Use of Knowledge.* SRI International, Artificial Intelligence Center, Menlo Park, Cal.

A Complex (Hungarian) Parser as an Embedded System

Balázs Kis, Mátyás Naszódi, Gábor Prószéky

MorphoLogic
{kis,naszodim,proszeky}@morphologic.hu

This paper presents the development, new features and applications of the HumorESK parser being developed since 1995.

HumorESK does not implement a fundamentally new parsing algorithm. It performs parsing in a bottom-up fashion; individual symbols are assigned simplified feature structures. While building the parse forest, it checks and propagates values in these feature structures. The logic of HumorESK operation closely resembles that of the PATR-II grammar formalism (Shieber, Uszkoreit, Pereira, Robinson, Tyson 1983).

The novelty in the HumorESK implementation is the method of applying rules and the grammar description itself. Rules are represented as finite sets of underspecified patterns, so that the system can use a lexicon similar to those in morphological analyzers. Each pattern consists of underspecified components (symbols); some symbols are described only by the label of the morphosyntactic class; others have the lexical form (the lemma) or even the surface form specified.

The author of the grammar is free to decide on the diversity of the morphosyntactic labels and the level of utilization of the feature structures. The architecture of HumorESK even enables to match rules without respect to the constituents' surface order – this feature is still under development. In the latter case, symbols can be derived to form a non-projective syntactic structure.

The HumorESK architecture has been designed to enable the parser to easily integrate with various applications as an embedded component. The program is entirely data-driven, which means that it makes no assumptions on the particular language, the grammar and the depth of parsing. This makes it possible to parse entire sentences or multiple-sentence structures to a great depth; this could drive detailed content extraction applications (the authors are able to present such an application as well). With other grammars, HumorESK can parse partial structures to a small depth, even perform shallow parsing; this provides for NP chunking or generalized collocation search. Parsing depth and scope can be different with the same grammar as well: grammars can be split into multiple levels, and the configuration can specify the last level HumorESK will execute, so parsing can be stopped before it reaches the topmost symbols.

It is an important feature of HumorESK that the patterns representing the rules can be assigned transformations. Thus, with appropriate formulation of the rules, the system can 'translate' the text into a logical structure in parse time.

HumorESK is a real-time system; parsing time can be limited; parsing data can be retrieved even when, according to the parsing logic, the process is not complete for a given segment.

In the presentation, the authors will briefly describe applications and projects where the HumorESK parser was utilized.

Magyar főnévi WordNet-ontológia létrehozása automatikus módszerekkel

Miháltz Márton

MorphoLogic Kft.
1118 Budapest, Késmárki utca 8.
mihaltz@morphologic.hu

Kivonat. A cikk bemutatja a folyamatban lévő, magyar főnévi WordNet adatbázis létrehozását célul kitűző munkálatok módszereit és legfrissebb eredményeit. Bemutatjuk azt a 9 különböző számítógépes módszert, melyek célja magyar főnevek automatizált hozzárendelése az angol nyelvű, 1.6-os verziójú WordNet synsetjeihez. A felhasznált magyar főnevek egy elektronikus magyar-angol kétnyelvű szótár szóanyagából származnak. A heurisztikus hozzárendelések támogatásához a kétnyelvű mellett az egynyelvű magyar Értelmező Kéziszótár számítógéppel feldolgozható anyagából nyertünk ki strukturális és szemantikai információkat. A különböző folyamatok eredményeinek pontosságát egy kézzel egyértelműsített etalon halmaz segítségével becsültük meg, majd a főnévi adatbázist a validált eredményhalmazok különböző szintű pontosságot meghaladó kombinációival állítottuk elő.

Kulcsszavak: WordNet-építés, Magyar Főnévi WordNet, automatikus szemantikai információ-szerzés

1 Bevezetés

Napjainkban az intelligens számítógépes nyelvészeti alkalmazások—természetes nyelvi szövegek gépi feldolgozását segítő eszközök, keresőmotorok, fordítóprogramok—fejlesztésében egyre inkább szükség mutatkozik természetes nyelvi fogalomtárak, ontológiák, lexikális tudásbázisok alkalmazására. Az egyik legelterjedtebb nyelvi ontológiai formalizmus a WordNet (WN), mely eredetileg a mentális lexikon számítógépes modelljeként született ([5]). A WN a tartalmas szóosztályok (főnevek, igék, melléknevek és határozószók) lexikai elemeinek szemantikai hálózata, ahol a fogalmi csomópontokat *synsetek*, szinonima-halmazok alkotják, közöttük olyan szemantikai kapcsolatokkal, min például a hipernima („Az-egy") reláció.

A magyar WordNet létrehozását megcélzó projekt 2000-ben indult, első lépésként a legkiterjedtebb tartalmas szóosztály, a főnevek adatbázisának létrehozását megjelölve ([6]). Munkánk során automatikus eljárásokat alkalmaztunk az ún. kiterjesztési módszer („Extend method", [8]) megvalósítására, melynek lényege, hogy a szabadon hozzáférhető, angol nyelvű Princeton WordNet főnévi synsetjeit megfeleltetjük magyar főnevekkel. Mivel feltételezzük, hogy a főnévi fogalmak mind az angol, mind a magyar nyelvben hasonló szemantikai rendszerbe szerveződnek, hiszen ugyanazt a vi-

lágot írják le, ezzel a módszerrel gyorsan és hatékonyan előállítható egy magyar főnévi ontológia kiinduló változata, mely szemantikai kapcsolatait az angol WN-től örökli.

2 A felhasznált számítógépes erőforrások

Az alapfeladat (magyar főnevek angol synsetekhez kapcsolása) megvalósításának kiinduló anyaga a MorphoLogic Kft. angol-magyar szótári adatbázisa volt, mintegy 17 700 magyar főnévi címszóval, melyekhez 12 400, az angol WN által lefedett angol fordítás tartozik.

Az illesztési folyamat támogatására felhasznált másik erőforrás az XML formátumba konvertált *Magyar Értelmező Kéziszótár* (ÉKSz, [3]) anyaga volt. Az ÉKSz mintegy 42 000 főnévi címszót tartalmaz, melyekhez több mint 64 000 különböző szöveges definíció tartozik.

3 Az alkalmazott módszerek

Mivel a kétnyelvű szótár magyar címszavainak nagy része egynél több (az egész szótár anyagában átlagosan 1,7) angol fordítással rendelkezik, az angol megfelelők pedig az angol WN-ben gyakran többértelműek, azaz egynél több (átlag 2,16) synsethez tartoznak, a megfeleltetés során egyértelműsíteni kellett, vagyis a lehetséges angol synsetek közül kiválasztani azokat, amelyekhez a magyar szó tartozik (jelenti őket). Ezt a feladatot automatikus módon, 9 különböző—részben korábbi, hasonló projektek során ([1], [2]), részben általunk kifejlesztett—heurisztikus eljárás alkalmazásával oldottuk meg. Ezzel a módszerrel a költséges manuális munka csupán az eredmények ellenőrzésére redukálódik.

3.1 A kétnyelvű szótár anyagával támogatott módszerek

A heurisztikák első csoportja a kétnyelvű szótárból kinyerhető információkon alapszik. Ezek egy része, melyeket a eredetileg a spanyol WordNet létrehozásakor fejlesztettek ki Atserias és munkatársai ([1]), a kétnyelvű szótár magyar és angol szavai, illetve az angol címszavak és a WN megfelelő synsetjei közötti kapcsolatokból kinyerhető információkat hasznosítják. A következő heurisztikákat alkalmaztuk:

- EGYJELENTÉSŰ FORDÍTÁSOK: ha egy magyar szó valamelyik angol fordítása egyértelmű a WN-ben, vagyis csupán egyetlen synsetbe tartozik, akkor létrehozunk egy kapcsolatot a magyar szó és a synset között.
- VARIÁNSOK: ha egy WN synset kettő vagy több olyan angol szót tartalmaz, melyeknek csupán egyetlen magyar fordításuk van, és az ugyanaz a magyar szó, akkor a magyar szót hozzárendeljük a közös synsethez.
- METSZET MÓDSZER: a magyar szavakat hozzárendeli azokhoz a synsetekhez, amelyek legalább kettőt tartalmaznak a szó angol fordításai közül.

Egy negyedik, általunk kifejlesztett heurisztika a kétnyelvű szótár magyar oldalából kinyerhető morfo-szemantikai információkon alapul. A magyar címszavak egy része termékeny endocentrikus (főnév+főnév) szóösszetétel Az ilyen összetételű szavak egy részének jellemző tulajdonsága, hogy a szerkezet alaptagja—az összetétel utótagja—többnyire meghatározza azt a szemantikai mezőt, amelynek az összetétel által jelölt dolog eleme ([4]). Így például a *hangversenyzongora* összetett szót számítógépes morfológiai elemzéssel felbontva a *hangverseny+zongora* morfémákra, az utótag kiválasztásával megkapjuk az összetett szó DERIVÁCIÓS HIPERNÍMÁJÁT („a *hangversenyzongora* az egy (fajta) *zongora*"). Ez az információ a következő részben leírt, módosított fogalmi távolság formula segítségével felhasználható a lehetséges angol synsetek feletti egyértelműsítéshez.

3.2 Az Értelmező Kéziszótár anyagát hasznosító módszerek

A Magyar Értelmező Kéziszótár főnévi definícióit a Humor morfológiai elemzőprogrammal ([6]) dolgoztuk fel, majd ez elemzett szövegben morfo-szintaktikai mintázatok heurisztikus keresésével ismertettünk fel szemantikai relációkat. Ezáltal képesek voltunk a címszóhoz 53 300 főnévi definícióban hipernímákat, 10 500 definícióban szinonimákat, illetve további 826 definícióban holonímákat és 584 esetben meronímákat azonosítani. Ezeknek a szemantikai információknak egy részét az alábbi módszerekkel használtuk fel a magyar címszavak WN-hez képest történő egyértelműsítéséhez:

- SZINONIMÁK: a magyar címszó angol fordításaihoz tartozó synsetek közül azt választjuk ki, amely a legtöbbet tartalmazza a szinonima angol fordításai közül (de legalább kettőt).
- HIPERNÍMÁK: azokban az esetekben, ahol mind a magyar címszónak, mind a hozzá azonosított hiperníma szónak volt angol fordítása a kétnyelvű szótárban, az 1. Ábrán bemutatott, módosított fogalmi távolság formula alkalamazásával választottuk ki a megfelelő angol synsetet. Az eredeti formulát Atserias és munkatársai fejlesztették ki ([1]).

$$dist'(w_1, w_2) = \min_{\substack{c_{1i} \in w1 \\ c_{2j} \in w2 \\ depth(c_{1i}) < depth(c_{2j})}} |path(c_{1i}, c_{2j})|$$

Fig. 1. A módosított fogalmi távolság formulát magyar főnévek és hipernímáik angol fordításainak párjaira alkalmaztuk. A képlet azt a két WN synsetet adja vissza, amely a WN hipernímahálózatában a legközelebb helyezkedik el egymáshoz. A magyar címszót a mélyebben lévő (a címszóhoz tartozó) synsethez rendeljük

Egy harmadik heurisztika a mintegy 1 500 ÉKSz címszóhoz megtalálható LATIN megfelelőket használja fel. Ezek általában állat- és növényfajok, rendszertani kategóriák, betegségek stb. latin nevei, melyek az angol WordNetben is megtalálhatók, így a latint egyértelműsítő közvetítőnyelvként felhasználva vihetjük végbe a hozzárendeléseket.

A kétnyelvű és az értelmező szótáron alapuló módszerek eredményeit az 1. Táblázat ismerteti.

Table 1. A különböző illesztési módszerek eredményei: illesztett magyar főnevek és WN synsetek, valamint a közöttük létrejött kapcsolatok számai

Módszer	Magyar főnevek	WN 1.6 synsetek	Kapcsolatok
Egyértelmű fordítások	8 387	5 369	9 917
Metszet módszer	2 258	2 335	3 590
Variáns módszer	164	180	180
DerivHip + FT	1 869	1 857	2 119
ÉKSz szinonimák	927	707	995
ÉKSz hipernímák + FT	5 432	6 294	9 724
ÉKSz latin megfelelők	1 697	838	848

3.3 Módszerek a lefedettség további növelésére

Azoknál a magyar főneveknél, ahol a magyar-angol szótár nem tartalmazott angol fordítást az ÉKSz alapján hozzájuk azonosított hiperníma vagy szinonima szavakhoz, két további módszerrel jutottunk angol fordítással rendelkező (derivációs) hipernímákhoz.

Az első módszer a 3.1 részben ismertetett eljárással, illetve termékeny főnév-főnév képzések felismerésével (pl. *ruhadarab* $\Rightarrow$ *ruha*) keres derivációs hipernímákat a szinonimákhoz és hipernímákhoz. Mivel a hiperníma-reláció tranzitív, a címszó hipernímájának (vagy szinonimájának) hipernímája is hipernímája lesz a címszónak.

A második eljárás kikeresi az azonosított hiperníma (vagy szinonima) szót az ÉKSz címszavai között, és amennyiben az egyértelmű (egyetlen definíciója van csak, tehát nincs szükség a jelentések közötti egyértelműsítésre), az ahhoz azonosított hiperníma szót használja fel (ha az rendelkezik angol fordítással).

Ezzel a két módszerrel 9,2%-os emelkedést tudtunk elérni az automatikusan illesztett magyar főnevek lefedettségében. Az automatikus módszerek összesen 13 948 magyar főnevet rendeltek hozzá 12 085 angol WN synsethez, 22 169 kapcsolatot létrehozva.

4. Az eredmények validációja és egyesítése

A különböző módszerek eltérő megbízhatóságúak, különböző pontosságú eredményeket produkálnak. Ezek pontos ellenőrzéséhez a kétnyelvű szótár teljes magyar oldalának anyagából véletlenszerűen kiválasztottunk 400 főnevet, melyek az angol fordításaikon keresztül összesen 2 201 lehetséges WN synsethez tartoznak. A lehetséges kapcsolatokat kézzel egyértelműsítettük, kiválasztva azokat, amelyek fennállnak és kitörölve azokat, amelyek nem állnak fenn a magyar szavak és az angol synsetek között. Ezzel a módszerrel létrehoztunk egy etalon halmazt, melynek segítségével elvégezhető a részeredmények megbízhatóságának becslése.

Elsőként megvizsgáltuk a 9 automatikus módszer eredményeit. Minden heurisztika esetében megállapítottuk a heurisztika és az etalon halmaz által közösen lefedett magyar szavak halmazát, az ezekhez a heuriszitka, illetve az etalon által rendelt kapcsolatokat, valamint ezeknek a kapcsolathalmazoknak a metszetét. Két mérőszámmal jellemeztük egy adott heurisztika megbízhatóságát. A pontosság (precision) érték a metszet halmaz és a heurisztika által létrehozott kapcsolathalmaz, a fedés (recall) érték pedig a metszet és az etalonban található kapcsolatok arányát jelzi. Az eredmények a 2. Táblázatban láthatók. Ebben azt is feltüntettük, hogy az adott módszer a kétnyelvű szótár teljes magyar oldalának milyen arányához rendelt kapcsolatokat (lefedettség (coverage) érték).

Table 2. Az etalon halmaz alapján számított pontosság és fedés értékek, valamint a kétnyelvű szótár magyar oldalának lefedettsége a különböző automatikus módszerek esetében, pontosság szerint csökkenő sorrendben. A latin ekvivalenseket felhasználó módszert nem tudtuk ezzel a módszerrel értékelni, mivel az jórészt szaknyelvi, az etalon halmaz általános szókincsében nem szereplő szavakhoz rendelt synseteket. Kézi mintavételezéssel és ellenőrzéssel ennek a módszernek a pontosságát kb. 80%-osra becsültük

Módszer	Pontosság	Fedés	Lefedettség
Variánsok	92.01%	50.00%	0.50%
Szinonimák	80.00%	39.44%	8.00%
DerivHip	70.31%	69.09%	17.50%
Lef. növ. 1.	67.65%	46.94%	7.50%
Egyért. ford.	65.15%	55.49%	69.25%
Metszet	58.56%	35.33%	17.50%
Lef. növ. 2.	58.06%	28.57%	6.00%
Hipernímák	48.55%	41.71%	49.25%

A különböző forrásokból származó eredmények egyesítésében a spanyol WN készítői által alkalmazotthoz hasonló módszert követtük ([1], [2]). Elsőként meghatároztunk két megbízhatósági küszöbértéket (70%, illetve 65%), majd egyesítettük azokat az eredményhalmazokat, amelyek az etalon halmaz segítségével végzett pontosságbecslések alapján elérték, vagy meghaladták ezeket a küszöbértékeket. Így létrejött két eredményhalmaz, körülbelül 70, illetve 65 százalékos becsült pontossággal.

Ezután létrehoztuk azoknak az eredmény-halmazoknak a páronkénti metszeteit, amelyek nem szerepeltek a fenti két halmazban, majd ezekre is elvégeztük a pontosságbecslést az etalon halmaz segítségével. A lehetséges 13 metszethalmaz közül 9 becsült pontosság-értéke lett 65 százalék, vagy annál magasabb (ebből 8 metszethalmaz 70% vagy magasabb becsült pontosságú). Ezeket a kiválasztott metszethalmazokat hozzáadtuk a két alaphalmazhoz, így tovább tudtuk növelni a lefedettséget anélkül, hogy a pontosság jelentősen csökkent volna. A dolog mögött az az elgondolás húzódik, hogy az alacsonyabb pontosságú módszerek is adhatnak a küszöbértéket meghaladó pontosságú eredményeket, amennyiben több külön forrás is megerősíti őket.

A két kiinduló halmaz, a kombinált metszethalmazok, valamint a két végleges halmaz adatai a 3. Táblázatban láthatók.

Table 3. A különböző eredmények kombinációiból előálló halmazokban található magyar szavak, angol synsetek és kapcsolataik száma, a halmazok becsült pontosságával

Eredményhalmaz	Szavak	Synsetek	Kapcsolatok	Pontosság
1. alaphalmaz	2 445	2 170	2 722	76,14%
További metszethalmazok	7 183	6 142	8 579	76,70%
1. végleges halmaz	**7 927**	**6 551**	**9 635**	**75,38%**
2. alaphalmaz	12 275	11 597	20 439	65,11%
További metszethalmazok	3 110	2 698	3 431	66,91%
2. végleges halmaz	**12 839**	**12 004**	**22 169**	**63,35%**

5. Összegzés, további munka

A magyar főnévi WordNet adatbázis kiinduló változatait különböző automatikus módszerek eredményeinek kombinációival állítottuk össze. Egy manuálisan létrehozott etalon halmaz segítségével becsült pontosságértékek alapján két, eltérő méretű és pontosságú halmazt hoztunk létre a további munka számára. A továbbiakban szeretnénk főként kézi munka alkalmazásával (a helytelen kapcsolatok kiszűrésével) növelni az eredmények megbízhatóságát, illetve tovább növelni a lefedett magyar szavak számát (a legpontosabbnak bizonyult heurisztikák alkalmazásával további kétnyelvű szótármodulokra).

Hivatkozások

1. Atserias, J., S., Climent, X., Farreres, G., Rigau, H., Rodríguez: Combining multiple methods for the automatic construction of multilingual WordNets. Proc. of Int. Conf. on Recent Advances in Natural Language Processing, Tzigov Chark (1997)
2. Farreres, X., G., Rigau, H., Rodriguez: Using WordNet for building Wordnets. Proc. of COLING/ACL Workshop on Usage of WordNet in Natural Language Processing Systems, Montreal (1998)
3. Juhász, J., I., Szőke, G. O. Nagy, M. Kovalovszky (szerk.): Magyar Értelmező Kéziszótár. Akadémiai Kiadó, Budapest (1972)
4. Kiefer, F.: Jelentéselmélet. Corvina, Budapest (2001)
5. Miller, G. A., R. Beckwith, C. Fellbaum, D. Gross, K. J. Miller: Introduction to WordNet: an on-line lexical database. Int. J. of Lexicography 3 (1990) 235–244.
6. Prószéky, Gábor: Humor: a Morphological System for Corpus Analysis. Language Resources and Language Technology, Tihany (1996) 149–158
7. Prószéky, G. M. Miháltz: Automatism and User Interaction: Building a Hungarian WordNet. Proc. of the Third International Conference on Language Resources and Evaluation, Las Pa mas de Gran Canaria, Spain (2002)
8. Vossen, P.: Right or Wrong. Combining lexical resources in the EuroWordNet project. Proceedings of Euralex-96, Goetheborg (1996)

Constructing a Nominal Hungarian WordNet Ontology with Automatic Methods

Márton Miháltz

MorphoLogic Kft.
1118 Budapest, Késmárki utca 8.
mihaltz@morphologic.hu

Keywords: WordNet construction, Hungarian Nominal WordNet, automatic extraction of semantic information

Abstract

This paper presents recent results of the ongoing project aimed at creating the nominal database of the Hungarian WordNet. We present 9 different automatic methods, developed for linking Hungarian nouns to WN 1.6 synsets. Nominal entries are obtained from two different machine-readable dictionaries, a bilingual English-Hungarian and an explanatory monolingual (Hungarian). The results are evaluated against a manually disambiguated test set. The final version of the nominal database is produced by combining the verified result sets and their intersections when confidence scores exceeded certain threshold values.

Our basic strategy was to attach Hungarian entries of a bilingual English-Hungarian dictionary to the nominal synsets of Princeton WordNet (WN), following the so-called extension approach. This way, the synsets formed by the Hungarian nouns can inherit the English WN semantic relations. In order to achieve this, we used heuristic methods, developed partly by previous similar projects and partly by us, which rely on information extracted from two machine-readable dictionaries (MRDs). This approach relies on the assumption that nominal conceptual hierarchies, which describe the world, would be similar across English and Hungarian languages to a degree which is sufficient for producing a preliminary version of our WordNet.

The first MRD we used was a bilingual English-Hungarian (17,700 Hungarian nominal entries, 12,400 English equivalents), serving as the basis of the attachment procedure. The *Magyar Értelmező Kéziszótár* (EKSz) monolingual explanatory dictionary (42,000 nominal headwords, 64,000 different definitions) was used to gain semantic information in order to assist the disambiguation heuristics

A number of these methods relied on structural information extracted from connections in the bilingual dictionary and WN (3 heuristics), and morpho-semantic information gained from the Hungarian side of the bilingual (1 heuristic). For the further support of the task we used the morphologically analyzed nominal definitions of the explanatory dictionary. The synonyms and hypernyms extracted this way were used by 4 additional heuristics. Further help was provided by the Latin translation equivalents available for a number of EKSz entries (1 heuristic).

In order to evaluate the precision of the results from the different information sources, we randomly selected 400 nouns from the Hungarian side of the bilingual and accomplished manual disambiguation against WN for these. By combining result sets whose precision scores were estimated with the help of this gold standard, we produced two preliminary versions of the Hungarian nominal database. The first version covers 7,900 nominal entries and 6,500 synsets (with 75% estimated precision), and the second, larger version covers 13,600 entries and 15,100 synsets (with 63% estimated precision).

Automatikus információszerzés gazdasági rövidhírekből

Prószéky Gábor

MorphoLogic
Budapest
proszeky@morphologic.hu

Kivonat. Az alábbiakban bemutatjuk azt a magyar nyelvre készített információkivonatoló rendszert, a *NewsPro*-t, melyet az NKFP 2/017/2001 projekt keretében készített a MorphoLogic Kft., az MTA Nyelvtudományi Intézetének Korpusznyelvészeti Osztálya, a Szegedi Egyetem Informatikai Tanszékcsoportja és a Gallup Intézet alkotta konzorcium. A kutató-fejlesztő munkát a gazdasági, ezen belül is elsősorban a céginformációs rövidhírek információtartalmának automatikus kinyerésére összpontosítottuk.

1 Tartalomelemzés és információkinyerés

Az üzleti élet gyakorlatilag minden területén a legfontosabb információk elektronikus szövegek formájában állnak rendelkezésre. Az információs társadalom legfontosabb eszközévé az interneten elérhető szövegek váltak. A gazdasági, társadalmi, politikai élet legfontosabb döntéseihez szükséges információk hozzáférési modellje az elmúlt 5-6 évben gyökeresen megváltozott. Mivel a döntés-előkészítéshez szükséges legfontosabb információk egyre inkább csak a számítógépes szövegfeldolgozás segítségével érhetők el, a pályázatunkban megcélzott feladatok megoldása stratégiai fontosságúvá vált. A dolgok természetéből adódóan az új protokollok által hálózaton keresztül is elérhető szövegek és egyéb adatok döntő többsége angol nyelven állt elő. A legfontosabb szövegfeltáró technológiákat is – érthető módon, az angol nyelv hatalmas piaci potenciáljának megfelelően – erre a nyelvre fejlesztik. Ugyanakkor az internet használói között minden előrejelzés szerint a következő években mind a tartalom, mind a felhasználók nyelve tekintetében más nyelvek kerülnek többségbe. A projektünkben kifejlesztett eszközök létfontosságúak ahhoz, hogy a mindennapi elektronikus üzleti tranzakciók és más információalapú szolgáltatások a nemzeti nyelveken folyhassanak. Olyan kis nyelv esetében, mint a magyar, az ilyen feladat mielőbbi megoldásának multiplikátorhatása van: technikai megoldásokat nyújtunk arra, hogy az elektronikus szövegtengerből információt, illetve – tartalomelemzési eljárások kidolgozásával – adatbázisokba tölthető tudást konvertáljunk.

A fenti célok megvalósítása jelentős mértékben igényelt nyelvtechnológiai alapkutatást, mivel a magyar nyelvű szövegek számítógépes feldolgozásának lehetőségei – bár az elmúlt években jelentős fejlődést mutattak – a projekt kezdetekor nem álltak azon a fokon, hogy megfelelhessenek az információkinyerés igényeinek. Az információkinyerés (Information Extraction, IE) az 1990-es években, az internet elterjedésével és az elektronikusan rögzített szövegek robbanásának hatására lett az információtechnológiai alkalmazások egyik legfontosabb céljává és eszközévé vált [1] [2]. Fontosságának megértéséhez azonban meg kell különböztetnünk az információkigyűjtéstől (Information Retrieval, IR) és a számítógépes nyelvmegértéstől (Natural Language Understanding, NLU). Az információkigyűjtés célja olyan dokumentumok (vagy más adatok) kikeresése, amelyek illeszkednek egy adott lekér-

dezésre, viszont nem célja a tartalom strukturált ábrázolása. A nyelvmegértés adott szöveg teljes tartalmának számítógépes ábrázolására irányul. Az ilyen eszközök a legkülönbözőbb területeken működnek, ám csak olyan szövegeket tudnak teljesen feldolgozni, amelyek megfelelnek valamely egyszerűsített nyelvtannak, és nem tartalmaznak a nyelvtan számára ismeretlen nyelvi elemeket.

Az információkinyerés célja strukturált – gépileg lekérdezhető, feldolgozható – adathalmaz előállítása szöveges dokumentumok tartalmából. Az így létrejövő adatbázis nem a szövegeket, hanem a belőlük kinyert releváns adatokat tartalmazza. Az információkinyerésnek azonban nem célja a teljes szövegtartalom ábrázolása. Az információkinyerés során nagy mennyiségű szövegből gyűjtünk ki információt. A folyamat során először minden szövegben meg kell keresni a releváns információt, azt strukturált formában ki kell vonni, majd azt egy előre meghatározott struktúrában tárolni. Lényeges, hogy az eljárás figyelmen kívül hagyja a nem releváns információt. Ezek a lépések egyrészt szigorúbban meghatározzák a feladatot, mint természetesnyelv-feldolgozás többi területe. Az elkészült rendszerek hatékonyságát pedig könnyű tesztelni az emberi és a számítógépes teljesítmény összehasonlításával. Az – angol nyelvű rendszereken kapott – teszteredmények szerint a legtöbb esetben legalább 20-30%-kal több és pontosabb információt nyújt a számítógépes információkinyerő rendszer, és ehhez lényegesen kevesebb időt igényel, mint az ember. Az információkinyerő rendszerek gyakorlati felhasználásai között megtalálhatjuk az adatbázis-építést nem rendezett szövegekből (példák: szakirodalom követése, üzleti és gazdasági problémák nyomon követése a sajtó rövidhírei alapján), összefoglaló rendszerek és a trendek megállapítására használható adatbányászati alkalmazások alapadatainak, valamint az információkigyűjtő rendszerekben használt indexek létrehozását.

Az információkinyerő rendszerek nem törekszenek a szövegek teljes elemzésére és megértésére, csupán a releváns részeket emelik ki és elemzik a részleges mondatelemzés eljárásával. Ez egyrészt növeli a sebességet, másrészt – a gyakorlati tapasztalat alapján – ez elegendő is az információ kinyeréséhez. Az ilyen rendszerek többek közt ennek köszönhetően rugalmasabbak is, mint a nyelvmegértő rendszerek, hiszen összetett és ismeretlen – és esetleg rosszul formált, nyelvtanilag helytelen – mondatokon is működik meghatározott – korlátozott – szemantikai területeken.

A természetes nyelvek kutatása informatikai szempontból a modern számítógépek számára is komoly kihívást jelent mind háttértár-kapacitásban, mind feldolgozási sebességben. Ennek oka a természetes nyelvekben előforduló jelentős mennyiségű szóalak, illetve a beszélt és írott nyelvben használt változatos – formális eszközökkel nagyon nehezen követhető – mondatstruktúra. Tovább bonyolítja a helyzetet a természetes nyelvekben előforduló ismeretlen tulajdonnevek kezelése, amely további erőfeszítéseket követel, ugyanis a meglevő számítógépes nyelvleírások a szinkron modellt követik, vagyis a nyelv valamely pillanatnyi állapotát írják le, zárt szótáraikkal együtt. Az informatika és az internet elterjedése miatt a természetes nyelvek interferenciája is nagy. Jelentős számú idegen – másik természetes nyelvből származó – elem épül be az egyes nyelvi szövegekbe. A hatékony informatikai rendszerek egyszerre több nyelvre is alkalmazható általános reprezentációt kell használjanak a feldolgozás különböző szintjein.

A kutatási erőfeszítések többsége jelenleg is az angol nyelvre összpontosul. Az információreprezentálásra irányuló magyar nyelvi kutatásokban eddig legfeljebb csak a kezdeti lépések történtek meg. Konzorciumunk tagjai közül többen vettek részt európai projektekben is, és az ott kidolgozott általános módszereket alkalmazták a magyar nyelvre is. A konzorcium e projekt keretében a magyar nyelvvel kapcsolatosan egyaránt folytatott alap- és alkalmazott kutatásokat. Reményeink szerint a projekt hosszú távra meghatározza a téma további kutatási irányait, mert olyan – a kutatások keretében referenciaként használható – szintaktikai és szemantikai kódolási technológiák, korpuszábrázolási modellek, definíciók készültek,

amelyek közös platformot teremtettek a további hasonló munkákhoz. A konzorcium további tevékenysége a projektben a magyar szövegek tartalmi feldolgozását előkészítő alap- és alkalmazott kutatásokra irányult. A célul kitűzött információkinyerési technológia kifejlesztéséhez olyan mondatelemző rendszer volt szükséges, amely nem törekszik feltétlenül szöveg mondatainak teljes elemzésére, azonban alkalmas arra, hogy a szövegben elszórtan szereplő – formálisan definiálható – szerkezeti elemeket felismerje, rendezze, kivonatolja, azaz információt vonjon ki belőlük. Ehhez kapcsolódó alapkutatási feladat volt a magyar nyelvben előforduló egyes (pl. főnévi) szerkezetek struktúrájának, szintaxisának vizsgálata is. Az utóbbinak feltétele volt, hogy feltárják a magyar nyelv szókincsének morfo-szintaktikai és egyes – korlátozott jelentéskörben érvényes – szemantikai jegyeit is. Alapkutatásunk többek közt tehát erre is irányult.

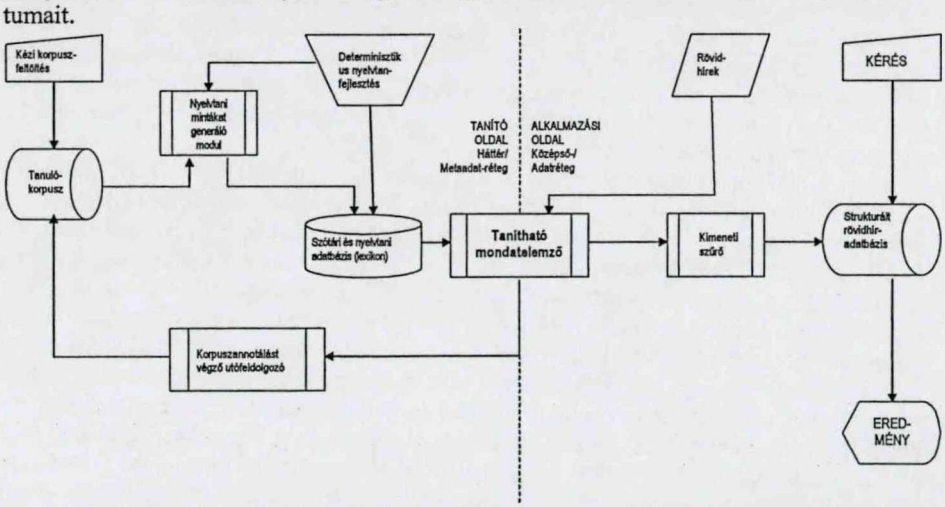
2 A kifejlesztett NewsPro rendszer

A *NewsPro* rendszer architektúrája az 1. ábrán látható. A kutató-fejlesztő munkát a gazdasági, üzleti, pénzügyi, piaci, ezen belül is elsősorban a céginformációs rövidhírek információtartalmának automatikus kinyerésére összpontosítottuk. Az összegyűjtött híreket egységes annotációval (tartalmi kódolással) láttuk el, a Reuters, az AFP vagy a UPI által is széles körűen használt NewsML-szabvány [3] szerint. Az első feladat olyan adatbázis létrehozása volt, amely a magyar nyelv törzsszókincsének minden eleméhez tartalmazza a sikeres nyelvi (elsősorban mondat- és szöveg-) elemzéshez szükséges morfológiai (szóalaktani), szintaktikai (szóhatáron túli nyelvtani) és szemantikai (jelentésleíró) információt. Az utóbbi esetben csak alapvető (szótári) információról lehet szó, hiszen a jelentés teljes ábrázolása az emberi tudás olyan részletességű ábrázolását jelentené, ami a tudomány mai állása szerint belátható időn belül nem valósítható meg. Az eredményül kapott adatbázis külön tartalmazza az igei vonzatkeretek, illetve a főnevek és a melléknevek leírását. Az adatbázis elemeit jól definiált jegyekkel láttuk el. Ezután kialakítottuk azt az infrastruktúrát, amely lehetővé teszi, hogy a program tetszőleges nyelvtant befogadjon, és ennek felhasználásával elvégezze szövegek részleges vagy teljes elemzését. A program semmilyen előfeltételezéssel nem él a nyelvtant illetően, vagyis teljesen nyelvfüggetlen, illetve adatvezérelt, egyedül a nyelvtan leírásának formátuma van megkötve.

A mondatelemzés alapjául szolgáló szabályrendszer (mintanyelvtan) kifejlesztéséhez megfelelően annotált gyakorlókorpusz szükséges, ezért elengedhetetlen fontosságú volt egy nyelvtani szempontból több szinten annotált korpusz kifejlesztése. A gyakorlókorpusz annotálása két szinten történt. Az annotáció első szintje az egyes szavak helyes szófaji kóddal való ellátását (morfológiai elemzését) foglalja magába. Ez az annotálás egy az SZTE és a MorphoLogic korábbi projektje (IKTA-027/2000) során kifejlesztett szófaji egyértelműsítő program segítségével történt [4]. A szófaji egyértelműsítés egyben a szintaktikai elemzés előfeltétele. Az elemzés második szintje a szavak, szószerkezetek mondatban betöltött szerepének meghatározását, ezen belül is a főnévi csoportok bejelölését, azaz szintaktikai annotálást jelenti. Az annotálást egy előelemző eszköz segítségével végeztük, majd ezt követően az erre a célra kifejlesztett eszköz felhasználásával félautomatikus módon, kézzel javítottuk. A fentiek alapján az annotált korpusz elemzési fák adatbázisának (treebank-nek) is tekinthető, s mint ilyen, a maga nemében az első a magyar nyelvterületen. Ez a munka előkészítése volt egy nagyobb, általános mondatelemzési adatbázisnak (IKTA-5/037/2002).

A mondatelemzésben a főnévi szerkezetek leírására kifejlesztendő szabályrendszer két pillérre épül. Egyrészt a mondatelemző program tanulási képességeinek felhasználásával, induktív logikai eljárásokkal tanuló-adatbázis alapján felismerési szabályokat állítottunk elő az egyes szintaktikai elemekhez. A program tanulási képességének rugalmassága abban

nyilvánul meg, hogy a mintanyelvtan működés közben folyamatosan bővíthető, mert a rendszer nem algoritmikus szabályokra, hanem mintaillesztésre épül. A másik módszer a hagyományos nyelvészeti kutatás volt, melynek keretében meghatároztuk a magyar nyelv olyan mondatelemeit, amelyek relevánsak a kifejlesztendő tartalomelemzési technológia szempontjából. Már a csak morfoszintaktikai (kb. szófaji) kódokkal ellátott korpuszban is alkalmazhatók egyszerű mintaillesztési algoritmusok, reguláris kifejezések segítségével írt nyelvtanok, melyek alapján a különböző mondatszerkezeti elemek azonosíthatók. Az annotált korpusz elkészítésének az volt a célja, hogy azt feldolgozva különféle gépi tanulási technikák alkalmazásával olyan tudásbázist állítsunk elő, ami megfelelő minőségű támogatást biztosít egy automatikus szintaktikai elemző számára. Az eredmények szintaktikai szabályok, amelyek a HumorESK mondatelemző program [5] meglevő szabályai közé megfelelő konverzióval beilleszthetők. Műhelyünkben végül is *HumorESK LangXtract* néven olyan információkivonatoló program keletkezett, amely a korábban kifejlesztett HumorESK mondatelemző programot felhasználva a bemeneti szövegben megjelöl meghatározott nyelvi elemeket. A mondatelemző program rendkívül részletes információt ad egy-egy szövegrész nyelvtani szerkezetéről. A nyelvi kivonatoló program szabadon konfigurálható a tekintetben, hogy az elemzőmodul által visszaadott információtömegből mely elemeket értékeljük relevánsnak. Így e program segítségével a tartalomelemzés szempontjából érdekes nyelvi elemeket úgy lehet kiemelni, hogy a feldolgozást nem zavarja az elemzés során létrehozott többi, a nyelvi struktúra köztes szintjeit felépítő szimbólum – vagyis a program elvégzi a válogatás jelentős részét. Ennek folytatásaként elkészült a *FrameTagger* nevű program prototípusa, mely egyrészt elvégzi a szövegben a HumorESK LangXtract által megjelölt névszói és más nyelvi szerkezetek szemantikai annotációját, továbbá absztrakt eseménymintákat (szemantikai kereteket) próbál meg a mondatokra illeszteni. Sikeres illeszkedés esetén automatikusan meghatározza az adott esemény szereplőit, körülményeit, attribútumait. A program az input állományban azokat a mondatokat, amelyekre sikeres illesztést tudott végrehajtani, XML-címkékkel jelöli meg, csakúgy, mint a felismert esemény szereplőit, attribútumait.



1. ábra. A rövidhír-feldolgozó NewsPro rendszer architektúrájának vázlata

3 Egy példa a NewsPro rendszer alkalmazására

Az elkészített prototípus-rendszer egy nagyobb témakör, az üzleti, azon belül is a céginformációs rövidhírek világában igazodik el elsősorban. Az elkészített prototípus fő célja annak megmutatása volt, hogy a kitűzött célt, az üzleti hírekben szereplő egyszerűbb események automatikus felismerését és a hírekben szereplő információk automatikus kinyerését meg lehet valósítani. Így tehát attól a szövegtől, hogy

Az Erste Bank 16,5 százalékkal növelte nyereségét 164,6 millió euróról 191,8 millió euróra.

Ennek NewsPro-feldolgozása pedig valahogy így fest XML-ben:

```
<root id="640" class="S-FULL" start="1" end="14">
<event schema="increased.midterm_report.income.1" roles_matched="6/6">
- <rv role="company" pos="N" case="NOM" sem="company|institute">
- <NP id="85" class="NP-FULL" start="1" end="3" head_lex="bank"
      HSK_head_lex="bank" case="NOM" ownernum="nil" ownerpers="nil"
      postp="" sem="company countable human institute">
    <w id="0" class="DET" at="1-1" lex="az" case="NOM">Az</w>
    <w id="2" class="UNKNOWN" at="2-2" lex="Erste">Erste</w>
    <w id="4" class="N" at="3-3" lex="bank" case="NOM">Bank</w>
  </NP>
  </rv>
- <rv role="_1" pos="V" lemma="növel">
    <w id="10" class="V" at="6-6" lex="növel">növelte</w>
  </rv>
- <rv role="trade" pos="N" lemma="eredmény|nyereség|profit" case="ACC"
      possessed_by="company" sem="abstract">
- <NP id="107" class="NP-FULL" start="7" end="7" head_lex="nyereség"
      HSK_head_lex="nyereség" case="ACC" ownernum="nil" ownerpers="nil"
      postp="" sem="abstract">
    <w id="12" class="N" at="7-7" lex="nyereség" case="ACC">nyereségét</w>
  </NP>
  </rv>
- <rv role="measure" pos="N" lemma="százalék" case="INS"
      modified_by_number="YES">
- <NP id="97" class="NP-FULL" start="4" end="5" head_lex="százalék"
      HSK_head_lex="16,5" case="INS" ownernum="nil" ownerpers="nil"
      postp="" sem="abstract countable">
    <w id="6" class="NUM" at="4-4" lex="16,5" case="NOM">16,5</w>
    <w id="8" class="N" at="5-5" lex="százalék" case="INS">százalékkal</w>
  </NP>
  </rv>
- <rv role="old_value" pos="N" case="DEL" modified_by_number="YES"
      sem="currency">
- <NP id="122" class="NP-FULL" start="8" end="10" head_lex="euró"
      HSK_head_lex="164,6" case="DEL" ownernum="nil" ownerpers="nil"
      postp="" sem="countable currency measure">
    <w id="15" class="NUM" at="8-8" lex="164,6" case="NOM">164,6</w>
    <w id="17" class="NUM" at="9-9" lex="millió" case="NOM">millió</w>
    <w id="19" class="N" at="10-10" lex="euró" case="DEL">euróról</w>
  </NP>
  </rv>
- <rv role="new_value" pos="N" case="SUB" modified_by_number="YES"
      sem="currency">
- <NP id="137" class="NP-FULL" start="11" end="13" head_lex="euró"
      HSK_head_lex="191,8" case="SUB" ownernum="nil" ownerpers="nil"
      postp="" sem="countable currency measure">
    <w id="21" class="NUM" at="11-11" lex="191,8" case="NOM">191,8</w>
    <w id="23" class="NUM" at="12-12" lex="millió" case="NOM">millió</w>
    <w id="25" class="N" at="13-13" lex="euró" case="SUB">euróra.</w>
  </NP>
  </rv>
</event>
```

Ez a kimenet nem olvasható könnyen „emberi" szemmel, de annyit észrevehetünk, hogy az információkivonatolás a nyelvtani elemzés eredményeire épül. A NewsPro rendszer a mondatban úgynevezett eseménysémákat (event schema) azonosított. Az eseménysséma meghatározza az esemény fajtáját és a résztvevőket. Ha tehát egy vállalat tulajdonosváltásáról van szó, akkor a rendszer „tudja", hogy itt meg kell nevezni a vevőt, az eladót, az adásvétel tárgyát, az árat (ha rendelkezésre áll), illetve a kérdéses tulajdoni hányadot. Ezek az elemzésben az „rv role" kezdetű sorokban láthatók. E struktúrát – vagy szűrés után egyes elemeit – kell adatbázisban tárolni. Ezt megkönnyíti, hogy a rendszer szabványos témabesorolást és esemény-, illetve szerep-megnevezéseket alkalmaz, a kimenet pedig a NewsML-szabványt követi. A könnyebb áttekinthetőség kedvéért azonban létrehoztuk a rendszer relációsadatbázis-szerű kimenetét is, melyet a fenti példán illusztrálva bemutatunk:

1. **increased.midterm_report.income.1** (6/6)

company	Erste Bank
_1	növel
trade	nyereség
measure	16,5 százalék
old_value	164,6 millió euró
new_value	191,8 millió euró

4 További fejlesztések, alkalmazások

Miután a NewsPro rendszert erre a célra kifejlesztett referenciakorpuszon validáltuk, ellenőriztük, a projekt végső eredménye egy olyan, kompakt, más rendszerekbe ágyazható keretrendszer lett, amely – megfelelő adatbázissal feltöltve – alkalmas arra, hogy rövid szöveges adatok tartalomelemzését felhasználó további alkalmazásokban is helyt álljon. Ahogyan az internet tovább terjed, a szöveges formában előálló adattömeg is várhatóan exponenciálisan növekszik tovább. A projekt által előállított megoldásnak a lényege részben az, hogy az intelligens tartalomelemzés – meghatározott területeknek megfelelő – szótárainak és részнyelvtanainak segítségével az egyes szövegek leírhatók a kötött, attribútumokkal rendelkező adatbázis-modell alapján. Az így „kitöltött" adatbázis-mezők azután a legkülönbözőbb kvalitatív, tér- és időbeli attribútumok mentén is lekérdezhetőkké válhatnak.

5 Hivatkozások

1. Cowie, J and Lehnert, W. Information Extraction. *Communications of the ACM*, 39(1) (1996)
2. Gaizauskas, R and Wilks, Y. Information Extraction: Beyond Document Retrieval. *Journal of Documentation* 54/1 (1998)
3. *Introduction to NewsML*: `www.newsml.org` (2003)
4. Alexin, Z. – J. Csirik – T. Gyimóthy – K. Bibok – Cs.Hatvani – G. Prószéky – L. Tihanyi. Manually Annotated Hungarian Corpus. In: Paroubek, P. (ed.) *Proceedings of the 10th Conference of the European Chapter of the Association for Computational Linguistics*, Vol.2., 53—56, Budapest (2003)
5. Prószéky, G.: Syntax As Meta-morphology. *Proceedings of COLING-96,* Vol.2, 1123–1126. Copenhagen, Denmark (1996)

Information Extraction from Short Business News Items

Gábor Prószéky

MorphoLogic
Budapest
proszeky@morphologic.hu

This paper presents a research project aimed at the extraction of structured information from business news items. Content analysis and extraction have traditional methods, mainly for the English language. As the texts in this project are in Hungarian, and the goal is to extract information to the maximum possible extent. Information extraction (IE) does not require deep parsing of the input. However, a content analysis system working with business news items must be able to identify essential business processes and activities, along with the participants, dates/times, and amounts of money involved. This, in our view, requires more than shallow parsing. Moreover, the texts – short business news items – have many extra-linguistic elements such as dates, times and currency amounts that must be properly interpreted within the textual context. The main goal of the project is to develop a system to analyse Hungarian-language business news items of up to 100 words, and provide structured data of business actions and processes. This will serve as a linguistic pre-processor for the purposes of a query system capable of providing information on actions of a particular company or institution, overall market trends etc. Dates, times, currency amounts must be treated numerically, and processed in the proper context. In order to keep the development process closely related to real-life applications, one of the members of the project consortium is a leading polling institute that validates and tests the content analysis scheme within their own applications.

The research project started in September 2001, and focuses on Hungarian-language news items. This type of texts presents three crucial problems that must be addressed using HLT modules:

(1) Besides the large complexity of the Hungarian morphology, the texts incorporate multiple levels of indirection when referring to an actor (a company, a person, a product, an authority etc.). This phenomenon is specific to the style of Hungarian-language business news items. Therefore, proper NP chunking is difficult, because highest level NP's tend to be very long (often over 10 words), and its internal structure must be revealed to some extent to correctly spot the head of the NP – which can be a multi-word named entity itself.

(2) For the purposes of this application, it is not sufficient to spot named entities: they must be properly identified and mapped to a database entity in order to resolve references. People must be identified either by their job titles or their names; multiple names of the same company must also be recognized. The project members agreed that time references and other numbers must be treated like named entities. Moreover, relative time references must be resolved to absolute dates, based on the release date of the news item under analysis. Here the conclusion is that a sophisticated named entity resolver (NER) module must be applied.

(3) Actions, subjects and objects, and their relationship must be properly parsed as each news item must be described by means of one or more strongly typed event description, which is then suitable to be included in a database. This implies that Hungarian verbal frames must be recognized by the system, as well as the relationship between multiple predicates within the news item, each represented as a single verbal frame.

Here the difficulty lies with the lack of a suitable grammar description of Hungarian. Therefore, a significant effort is required to perform basic research on the Hungarian grammar – all as part of this IE project.

Part of the problem is that the text of news items tends to be ill-formed. Thus the grammar model for the IE application will not be a 'pure' description for the Hungarian grammar, as content must still be extracted when the text does not conform to Hungarian spelling rules or grammar/style conventions. Here it is not necessary to spot the errors: the system must only return extracted information, 'pretending' the input was correct.

Nyelvészeti tudásforrások integrálási lehetőségei diszkriminatív szegmens-alapú beszédfelismerő rendszerekbe

Tóth László,[1] Kocsor András,[2] Kovács Kornél,[3] Felföldi László [4]

MTA-SZTE Mesterséges Intelligencia Tanszéki Kutatócsoport,
H-6720 Szeged, Aradi vértanúk tere 1., Hungary
{[1]tothl, [2]kocsor, [3]kkornel, [3]lfelfold}@inf.u-szeged.hu
http://www.inf.u-szeged.hu/speech

Kivonat A gépi beszédfelismerésben jelenleg kizárólag csak statisztikai elven működő algoritmusokat használnak. Ezek egyszerű matematikai modelleken alapulnak, amelyek paramétereiket hatalmas adatbázisokon automatikusan hangolják be. Az algoritmikai szempontok sajnos háttérbe szorítják a fonetikai/nyelvészeti ismereteket, így ezek a modellek irreális egyszerűsítő feltevéssekel élnek a beszédkommunikáció természetére nézve. Egy lehetséges alternatíva az ún. szegmentális modellek használata, amelyek – a statisztikai alapelv feladása nélkül – enyhébb megszorításokra épülnek. Ebben a cikkben bemutatjuk a tanszékünkön fejlesztett OASIS szegmens-alapú felismerőt, amely diszkriminatív elven, azaz posteriori valószínűségek összekombinálásával dolgozik. Ennek további előnye, hogy nagyobb rugalmasságot biztosít a különféle szintű (de továbbra is statisztikai jellegű) nyelvi információk integrálására, mint a hagyományos rejtett Markov-modell.

1. Bevezetés

"Mivel járult hozzá a fonetika a gépi beszédfelismeréshez?" - tette fel a kérdést nemrégiben az utóbbi terület egyik neves képviselője egy konferencián. A jelenlegi felismerők ugyanis nem fonetikai ismeretekre, hanem statisztikai elvekre épülnek. Ezek hatékonyságukat annak köszönhetik, hogy paramétereiket hatalmas méretű tanító adatbázisokon optimalizálhatjuk. Ennek ára, hogy ehhez viszonylag egyszerű (s így irreális feltételezéseken alapuló) matematikai modellt kell készítenünk. Ezért a jelenleg legelterjedtebb technológia, a rejtett Markov modellezés (HMM) [4] számos egyszerűsítő feltevéssel él a beszédjel információkódolási módjára nézve. A statisztikai megközelítés azonban nem zárja ki eredendően a fonetikai vagy percepciós ismeretek beépítését: ezeket a modell struktúrájának kialakításakor használhatjuk fel (ún. inductive bias). A HMM problémáinak egy részén túlléphetünk az általunk is használt ún. szegmentális modellekkel. Ezen túlmenően az általunk alkalmazott diszkriminatív modellezési technika egyszerű módot kínál a magasabb szintű (statisztikai jellegű) nyelvi tudásforrásoknak a felismerésbe történő integrálására. Cikkünkben bemutatjuk a tanszékünkön fejlesztett OASIS rendszer jelenlegi felépítését, majd megvizsgáljuk, hogy a diszkriminatív szegmens-alapú séma milyen lehetőségeket kínál a különböző szintű nyelvi modellek beépítésére.

2. Beszéd-dekódolás az OASIS rendszerben

A gépi beszédfelismerés célja egy A beszédjelhez a hozzá legjobban illeszkedő $F = f_1...f_N$ szimbólumsorozat megtalálása, ahol az f_n-ek beszédhangokat, vagy valamely más, alkalmasan megválasztott kódolási egységeket jelölnek. Jóformán minden felismerési algoritmus feltételezi, hogy mindegyik szimbólumnak megfeleltethető a jel egy jól meghatározott A_n időintervalluma, azaz a jel fonetikailag szegmentálható (ez az első, jelen esetben fonetikai jellegű nem teljesen reális feltevésünk a dekódolás során). Mivel a jelnek sem a helyes S szegmentálása, sem a valódi F fonetikai átirata nem ismert, ezért a felismerés során végig kell vizsgálni a jel összes lehetséges szegmentálását és az azokra illeszthető összes lehetséges szimbólumsorozatot. A feldolgozás során az összes lehetőséghez valamilyen költség-értéket rendelünk, és a legjobb költségű esetet fogjuk a felismerés eredményének tekinteni. Algoritmikailag az esetek végignézése egy keresési feladatot jelent, amelyet szaknyelven dekódolásnak nevezünk. Erre a célra az általunk fejlesztett OASIS rendszerben olyan algoritmust akartunk adni, amely kellően rugalmas ahhoz, hogy sokfajta stratégiát ki lehessen próbálni. Jelen pillanatban a rendszer az alábbi általános sémát használja.

Legyen az A jel valamilyen $a_1...a_T$ akusztikus események sorozataként adott. A keresés során balról jobbra haladva beszédhangokat igyekszünk ráilleszteni a mérésekre, minden lehetséges szegmenshatárt figyelembe véve. A művelet során megoldáskezdemények serege áll elő, ezeket hipotéziseknek fogjuk nevezni. Egy $H(t, F, w)$ hipotézis minimálisan a következőket tartalmazza: t az az időindex, ameddig az A feldolgozásában eljutottunk, $F = f_1...f_t$ a jel eddigi részére illesztett szimbólumsorozat, w pedig az illesztéshez rendelt költség. A dekódolás vázlatosan a következő: kiválasztunk egyet az eddigi hipotézisek közül, és kiterjesztjük. Ez abból áll, hogy az eddigi végponttól előre "tapogatózunk" valahány mérési adatnyit, és megvizsgáljuk, hogy ezek milyen eséllyel (költséggel) képezhetnek egy következő beszédhangot. Mivel nem tudjuk, hogy pontosan hány mérési adat tartozik a következő hanghoz, ezért minden lehetőségből egy-egy újabb hipotézist képezünk. Az új hipotézisek költségét úgy kapjuk meg, hogy az eddigi költséget és az újabb hang illesztésének költségét összekombináljuk egy megfelelő függvény segítségével. A túloldali pszeudo-kód részletezi az algoritmust.

A hipotézistér bejárására sokféle stratégia létezik, például az időszinkron bejárás, vagy a veremalapú dekódolás és variánsai [4]. Az algoritmus akkor ér véget, ha találtunk egy megoldást, vagy már nincs több kiterjeszthető hipotézis. Többnyire beérjük az előbbivel, azaz csak a legelső illeszkedő szimbólumsorozatot keressük meg. Megfelelően megválasztott bejárási stratégiával ugyanis garantálni (de legalábbis jó eséllyel biztosítani) lehet, hogy az első megoldás egyben optimális is. Az algoritmus esetleg úgy is véget érhet, hogy nem talál illeszkedő sorozatot. Ezt nagymértékben befolyásolja a vágási feltétel, amelynek segítségével elvethetjük az esélytelennek látszó hipotéziseket. Ezzel jelentősen csökkenthetjük a keresési teret, de esetleg a jó megoldás kidobását is kockáztathatjuk. A kiterjesztés új beszédhangjába bevont akusztikus események számát a megállási feltétel limitálja. Erre legegyszerűbb megoldás korlátot adni a beszédhangok lehetséges hosszára. De ha w_f az események számának monoton függvénye, akkor egy w_f-re adott küszöb is használható. Végezetül, a költségszámítást a g_1 és g_2 függvények végzik. Előbbi az egyes beszédhangok, utóbbi a teljes hipotézis költségét méri, és természetesen mindkettő kulcsfontossággal bír a helyes hipotézis megtalálásában.

Algorithm 1 Általánosított beszéd-dekódolási algoritmus

megoldáslista := $\emptyset$
hipotézislista := $h_0(t_0, "", 0)$
while van kiterjeszthető hipotézis **do**
  válasszunk egy $H(t, F, w)$ kiterjeszthető hipotézist valamely stratégia alapján
  if $t = T$ **then**
    if csak az első megoldás kell **then**
      return H
    else
      helyezzük át H-t a megoldások közé
    end if
  end if
  for $t' = t+1, t+2 \cdots$ **do**
    for all f **do**
      $w_f := g_1(f, < t, t' >)$ ahol g_1 az $f < t, t' >$-re való illeszkedési költsége
      $w' := g_2(w, w_f)$ ahol g_2 egy megfelelő aggregációs függvény
      if vágási-feltétel(w_f, w') **then**
        képezzünk egy új $H'(t', Ff, w')$ hipotézist és tegyük a hipotézislistába
      end if
    end for
    if megállási-feltétel($< t, t' >$) **then**
      break
    end if
  end for
end while

2.1. Speciális eset: Rejtett Markov-modell

A fenti dekódolási sémában elvileg sokféle módon lehetne a költségeket meghatározó g_1 és g_2 függvényeket megválasztani. A gyakorlatban azonban statisztikai módszereket használnak, azaz a költségek valószínűségi értékeknek felelnek meg. Ennek oka, hogy az ún. Bayes döntési elv alapján optimális működés (minimális számú tévesztés) garantálható [4]. Ehhez azonban szükséges egy jó becslés a $P(F|A)$ valószínűségre nézve. A gépi tanulás számos módszert ismer a valószínűségek példák (tanító adatbázisok) alapján történő becslésére, a beszédfelismerési probléma azonban speciális, amennyiben a lehetséges A beszédjelek és F átiratok száma potenciálisan végtelen. Erre az egyetlen megoldás mindkettőt kisebb egységekre bontani. Ekkor kerül képbe a már említett szegmentális feltevés, azaz a jelet szegmentumokra bontjuk, és $P(F|A)$-t a szegmentumokhoz rendelt $P(f_n|A_n)$ értékekből kombináljuk össze. Erre a valószínűségszámítás lényegében egyetlen egyszerű módot kínál: ha az egységek függetlenek, akkor a valószínűségek összeszorozhatók. Ez a másik – ez esetben matematikai jellegű – olyan egyszerűsítő feltevés, amellyel élni fogunk, habár valójában nem teljesül.

Dekódolási sémánkba belefér a rejtett Markov-modellezésnek az az esete, amikor a modell szigorúan balról-jobbra típusú, és állapotok átugrása nem lehetséges. Folytonos beszéd felismerésére legtöbbször ilyet szoktak használni, három állapottal, ahol az állapotok (nálunk az f_n-ek) beszédhang-harmadoknak (kezdő-átmeneti, stabil, záró-átmeneti szakasz) felelnek meg [4]. A dekódolás során tehát beszédhangok helyett eze-

ket kell használnunk fonetikai szimbólumként. Az akusztikai megfigyeléseket 10-30 ezredmásodpercenként (szakszóval "keretenként") számolt a_l spektrális adatok képezik. A g_2 aggregációs függvény egyszerűen csak összeszorozza az egyes szimbólumokhoz rendelt értékeket. A lényeg a g_1 függvény, amely az alábbi alakot ölti:

$$g1(f, < t, t' >) = P(f|F) \cdot l_f^{(t'-t)} \cdot \prod_{i=t}^{t'} P(a_i|f), \quad (1)$$

ahol a harmadik tényező a $< t, t' >$ intervallum minden a_l méréséhez kiszámol egy $P(a_l|f)$ valószínűséget, és – függetlenséget feltételezve – ezeket összeszorozza. A második tényező egy exponenciálisan lecsengő hosszmodell, amely egy megfelelően beállított l_f konstanst igényel. Végezetül $P(f|F)$ a nyelvi modell hozzájárulása. A második két tényező ilyen módon való felírásának technikai okai vannak: mivel a különböző $< t, t' >$ intervallumokhoz rendelt költségek egymást tartalmazzák, így megfelelő technikával (dinamikus programozás) párhuzamosan, s így gyorsan számíthatóak. Egy további, talán még fontosabb szempont, hogy a modell komponenseit képező $P(a_l|f)$ eloszlások és l_f konstansok adatbázisok alapján történő tanulására hatékony algoritmusokat lehet adni (a $P(f|F)$ nyelvi modellt ezektől függetlenül szokás tanítani).

2.2. Speciális eset: Szegmentális modellek

Az adott beszédhanghoz sorolt $P(a_l|f)$ értékek összeszorzása hatékony ugyan, de a függetlenségi feltevés erősen irreálisnak tűnik. A hosszt leíró komponens exponenciális lecsengése sem felel meg a gyakorlati méréseknek. Ezen problémák feloldására javasolták az ún. szegmentális modellek használatát, amelyek a $< t, t' >$ szegmenst „egyben" modellezik [9]. Ennek lényege, hogy a HMM-nél látott $g_1(f, < t, t' >)$ számítási képlet második két tényezőjét lecseréljük valamilyen műveletigényesebb, de az adatoknak remélhetőleg jobban megfelelő modellre. A bonyolultabb megoldások parametrikus modelleket illesztenek a $< t, t' >$-hez tartozó a_l adatokra [9]. Az egyszerűbb megoldás először is $< t, t' >$-t annak hosszától függetlenül ugyanannyi adattal próbálja meg leírni. Ennek legkönnyebb módja az adatokat elsimítani, és fix számú mintát venni belőle [3]. Értelme pedig az, hogy az így kapott reprezentáción immár alkalmazható az a rengeteg fajta modellezési technika, amelyet a gépi tanulásban valaha felvetettek rögzített dimenziószámú terekben való osztályozásra, illetve valószínűségi regresszióra.

Egy további szempont is felvetődik itt, mégpedig az, hogy a rejtett Markov modell az ún. generatív modellek családjába tartozik. Ez azt jelenti, hogy a $P(a_l|f)$ valószínűségekből építkezik, szemben az ún. diszkriminatív modellekkel, amelyek az f szimbólumok $P(f|a_l)$ a posteriori valószínűségével dolgoznak. Ennek két okból van jelentősége: az egyik, hogy tapasztalatok szerint a diszkriminatív modellek kicsit jobb osztályozási eredményeket képesek elérni (bonyolultabb tanítási folyamat árán), mint a generatívak. A másik, hogy amennyiben a felismerés során többféle tudásforrást akarunk kombinálni, akkor erre a diszkriminatív modellezés sokkal többféle módot és lehetőséget kínál.

Az OASIS rendszerben az utóbbi években számos algoritmust kipróbáltunk beszédhangok diszkriminatív szegmens-alapú osztályozására. Az irodalommal összhangban mi is azt találtuk, hogy ez az egyszerű séma valamivel jobb beszédhang-felismerési eredményekre képes, mint a keret-alapú modellezés [7] [8] [11].

3. Nyelvi modellezés az OASIS jelenlegi verziójában

Mint láthattuk, a jelenlegi beszédfelismerési technika alapvetően valószínűségi alapú, és a nyelvi modelltől is azt várja, hogy valószínűségeket rendeljen a nyelvi egységekhez. Természetesen a hagyományos szabályalapú nyelvleírások is tekinthetők ilyennek, hiszen felfoghatók úgy, mintha kizárólag 0 és 1 valószínűségeket adnának ki. A gyakorlatban az ilyen modellek azonban túl merevnek bizonyultak, így érdemesebb az engedékenyebb valószínűségi modellekhez fordulni. Szerencsére a szabályalapú technikák közül több kiterjeszthető valószínűségi jellegűvé, így kaphatjuk például a sztochasztikus környezetfüggetlen nyelvtanokat (P-CFG) vagy a súlyozott automatákat.

Az OASIS rendszer nyelvi moduljának megtervezésekor igyekeztünk követni a más felismerőkben definiált nyelvleírási technikákat. Ehhez a Microsoft Speech API-ból indultunk ki, amelyben környezetfüggetlen nyelvtanokat lehet definiálni egy XML leírási formátumot követve. Mivel a SAPI-t angol nyelvre találták ki, így a nyelvtanok nemterminálisai közvetlenül a nyelv szavai. A magyar nyelv esetében viszont az összes lehetséges toldalékolt alak felsorolása kezelhetetlen. Szerencsére a magyar morfológia modellezése véges állapotú automatákkal jól megoldható [2]. Tapasztalatunk szerint egy adott szó toldalékolt alakjai automatával nagyságrendekkel kisebb helyen tárolhatóak, mint bármilyen hagyományos tömörítőprogrammal. Ezért a SAPI leírást kiterjesztettük oly módon, hogy nálunk a terminálisok helyére automatákat is be lehet ágyazni. Ez egy olyan környezetfüggetlen nyelvtanhoz vezet, amelynek terminális szimbólumai az automaták által felismert nyelv szavai. További tömörítést érhetünk el a morfológiát kezelő automaták tömör reprezentációjával. Ehhez speciális automatatömörítő algoritmusokat használunk, amelyek az adott nyelvet felismerő automaták közül a lehető legkisebbet konstruálják meg [5]. További tárcsökkentést jelent, ha az automatát is kis helyigényű adatszerkezettel, például a [6]-ban megadott módszer szerint tároljuk.

A valószínűségek kezelésére a Speech API-ban súlyokat rendelhetünk a szabályok alternatív jobboldalaihoz. Az OASIS nyelvi moduljában kiegészítésként bevezetett automaták szintén megengedik az egyes elágazások súlyozását, így a két szint kombinálásával a rendszer képes az egyes beszédhang-sorozatokhoz valószínűségeket rendelni.

A nyelvi modell interfészének kialakításánál figyelembe kellett vennünk, hogy a felismerést végző (azaz az 1. algoritmust végrehajtó) modul milyen formában várja a nyelvi modell támogatását. Mivel a hipotézisek kiterjesztése során egy adott hangsorozat lehetséges folytatásaira van szükségünk, ezért a nyelvi modul feladata egy adott prefix összes lehetséges folytatásait (azaz a következő beszédhangot) visszaadni. Így a nyelvi modul interfésze az alábbi két függvényből áll, amelyek iterátor-jellegű bejárást biztosítanak a nyelvi modell összes lehetséges beszédhang-sorozatának végignézéséhez:

`Enter`: A megadott prefixhez meghatározza az első lehetséges kiegészítést, és annak valószínűségét (ha nincs ilyen, akkor null-t ad vissza).

`Next`: Megadja (az ugyanazon prefixhez tartozó) következő lehetséges kiegészítést, és annak valószínűségét. Ha nincs több, akkor null-t ad vissza.

Technikai szempontból a modell automatáinak implementálása illetve bejárása viszonylag egyszerűen megoldható. A környezetfüggetlen nyelvtan kezelése azonban már veremautomata használatát igényli. Ebben az esetben a verem aktuális értékeinek tárolása is szükséges, ami a nyelvi modul megvalósítását megnehezíti.

4. További lehetséges nyelvészeti tudásforrások integrálása

Az 1. algoritmusban leírt dekódolási séma kellően általános, így könnyen kiterjeszthető nagyobb számú információforrás egyesítésére. Ehhez csak a g_1 (és esetleg a g_2) függvény(eke)t kell megfelelően módosítani. Gyakorlati szempontból a legkritikusabb pont, hogy a modellt ne bonyolítsuk el annyira, hogy az optimális paraméterek algoritmikus megtalálása lehetetlenné váljék. Szerencsére sok olyan matematikai módszer ismert, amely tudásforrások optimális integrálásáról szól, illetve az osztályozók kombinálása is rendkívül aktív kutatási téma az utóbbi időben. Továbbá a beszédfelismerésben egyre inkább terjednek az olyan optimalizálási módszerek, amelyek az általunk is használt diszkriminatív modellezést támogatják [10]. Ilyen például az ún. diszkriminatív modell-kombinálási technika, mellyel az alábbi típusú integrálást optimalizálhatjuk [1]:

$$P(F|A, L_1, ..., L_r) \approx \max_S \prod_i P(f_i|A, S)^{\alpha_0} P(f_i|L_1)^{\alpha_1} \cdots P(f_i|L_r)^{\alpha_r}, \quad (2)$$

ahol ez esetben r darab (nyelvi) információforrásunk van, $L_1, ..., L_r$, és ezek posterior valószínűségek formájában „szavaznak" az egyes f_i szimbólumokra. A források kombinálása hatványozás, majd szorzás útján történik. Természetesen másfajta kombinációval is próbálkozhatunk, de az optimális kombinálás megtalálása más esetekben más matematikai elveket kívánhat. Az OASIS rendszerben jelen állapotban még csak egyetlen nyelvi modell van (az előző fejezetben leírtaknak megfelelően), és kombinálási szabályként a hagyományos rendszerekben már bevált szorzást alkalmazzuk, azonban többfajta alternatív kombinálási technika kipróbálását is tervezzük a közeljövőben.

Hivatkozások

1. P. Beyerlein, Discriminative Model Combination, Proc. ICASSP'98, pp. 481-484., 1998.
2. Futó Iván (szerk.), Mesterséges intelligencia, Aula, 1999.
3. J. R. Glass, A probabilistic framework for feature-based speech recognition, Proc. ICSLP'96, pp. 2277-2280, 1996.
4. X. D. Huang, A. Acero és H-W. Hon, Spoken language processing, Prentice Hall, 2001.
5. Kertész-Farkas Attila, Fülöp Zoltán, Kocsor András: Magyar nyelvű szótárak tömör reprezentációja nemdeterminisztikus automatákkal, Ugyanebben a kiadványban
6. G. A. Kiraz, Compressed Storage of Sparse Finite-State Transducers, *Proc. of WIA'99 (Szerk. O. Boldt és H. Jürgensen)*, LNCS Vol. 2214, pp. 109-122, Springer, 2001.
7. Kocsor A. et al., A Comparative Study of Several Feature Space Transformation and Learning Methods for Phoneme Classification, Int. J. Speech Technology, Vol. 3, 3/4, pp. 263-276, 2000.
8. Kocsor, A. és Toth, L., Application of Kernel-Based Feature Space Transformations and Learning Methods to Phoneme Classification, elfogadva az Applied Intelligence folyóiratba
9. M. Ostendorf, V. Digalakis és O. A. Kimball, From HMMs to Segment Models: A Unified View of Stochastic Modeling for Speech Recognition, IEEE Trans. ASSP, 4:360–378., 1996.
10. R. Schlüter et al., Comparison of discriminative training criteria and optimization methods for speech recognition, Speech Communication, Vol. 34., pp. 287-310., 2001.
11. Tóth L., Kocsor A. és Kovács K.: A Discriminative Segmental Speech Model and its Application to Hungarian Number Recognition, Proceedings of TSD'2000 szerk. P. Sojka, I. Kopecek és K. Pala, LNAI 1902, pp. 307-313, Springer Verlag, 2000.

On the Integration of Linguistic Knowledge Sources in Discriminative Segment-Based Speech Recognizers

László Tóth,[1] András Kocsor,[2] Kornél Kovács,[3] László Felföldi [4]

Research Group on Artificial Intelligence, Hungarian Academy of Sciences
Aradi vértanúk tere 1., H-6720 Szeged, Hungary
{[1]tothl, [2]kocsor, [3]kkornel, [3]lfelfold}@inf.u-szeged.hu
http://www.inf.u-szeged.hu/speech

"What was the contribution of phonetics to automatic speech recognition?" This question was posed by a prominent researcher of the latter field in a recent conference. The fact is that current speech recognizers make use of practically no phonetic knowledge. In the early experiments with speech recognition there were attempts to create knowledge-based systems, but statistical methods soon took over. The efficiency of these is rooted in the fact that their parameters can be automatically optimized on huge training databases. But the price is that the optimization requires a very simple mathematical model – usually based on unrealistic or oversimplistic assumptions. Statistical methods, however, do not automatically exclude the incorporation of phonetic or speech perception knowledge. These can be taken into consideration when designing the structure of the model (which leads to the so-called inductive bias). The currently most popular modeling technique, Hidden Markov Modeling has several incorrect simplifying assumptions regarding the information coding nature of speech. Some of these restrictions can be alleviated by using the so-called segment-based models. The OASIS recognizer developed at our institute is also segment-based and, in accordance with the literature, we have also found that these models indeed give a better representation of phones than the traditional HMM technique. Moreover, the discriminative modeling scheme applied in our system provides an easy way of integrating higher-level (statistical) linguistic knowledge sources into the recognition process. In our paper we present the current structure the OASIS system, and examine what possibilities this scheme provides for the integration of linguistic knowledge sources.

Hangátmenetek a beszédfelismerésben

Sejtes Györgyi – Zsigri Gyula

Szegedi Tudományegyetem
sejtes@hung.u-szeged.hu, zsigri@hung.u-szeged.hu

Abstract. A beszédhangok többsége jól felismerhető a nekik tulajdonított szakasz közepéről, de ez nem mondható el a zárhangokról. Az alábbiakban a zárhangok kétféle szegmentálását hasonlítjuk össze. Az egyik megőrzi a felismerési mechanizmus egységességét, de az anyanyelvi beszélők intuíciójától eltérően a zárhangok zárrészét és felpattanását két különböző szegmentumként elemzi. A másik bonyolultabb felismerőmechanizmust igényel, de jobban modellálja a beszélők intuícióját, és a fel nem pattanó zárhangok felismerését is lehetővé teszi.

A beszédfelismerő programok úgy jutnak el oda, hogy a bejövő hanghullámokat képesek legyenek beszédhangok sorozataként elemezni, hogy előtte rengeteg olyan digitalizált hangfelvételt bocsátunk a rendelkezésükre, amelyekben emberek által beiktatott határjelölők osztják a hangfolyamot diszkrét szakaszokra, és az így kijelölt szakaszok mindegyikéhez egy-egy fonetikus jel van hozzárendelve. A határjelölők a hangfolyam természetéből következően csak körülbelüliek lehetnek. A beszédhangok nem úgy követik egymást, hogy egyszer csak véget ér az egyik, és csak azután kezdődik a másik, hanem úgy, hogy az egyik még tart, amikor a másik már elkezdődik. A határjelölők nem is arra valók, hogy pontosan megadjuk, hogy melyik beszédhang mettől meddig tart, hanem arra, hogy tudassuk a programmal, hogy hány beszédhangot kell felismernie, és hogy melyiknek melyik szakaszon belül keresse az azonosító jegyeit. A magánhangzók és a folyamatosan hangoztatható mássalhangzók széleit a program levágja, és a középső rész alapján próbálja kiszűrni, hogy melyik beszédhangot miről lehet felismerni. Ez a módszer azonban a zárhangokkal már nem működik.

Papp István (1966: 86), sok más szerzőhöz hasonlóan, a zárhangok képzésének három egymás utáni mozzanatát különbözteti meg:

a) a záralkotást (implosio): kezdő mozzanat
b) a zár tartamát (occlusio): középső mozzanat
c) a zár megszüntetését (explosio): befejező mozzanat

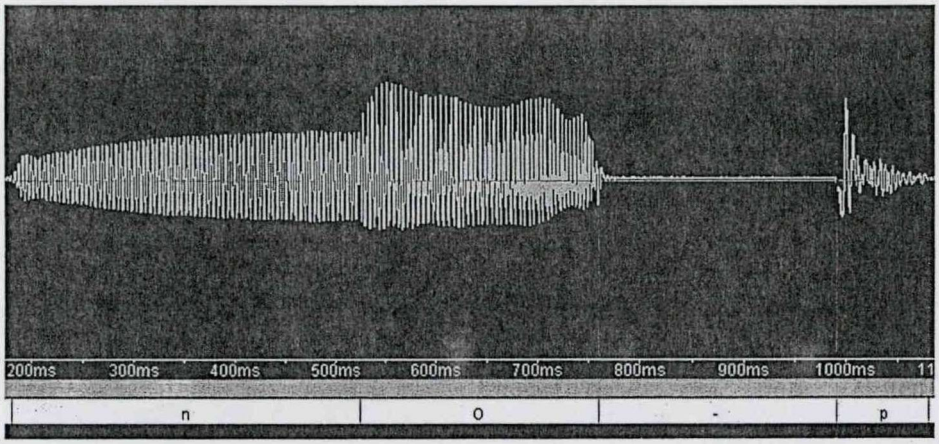
A zöngétlen zárhangok középső mozzanata teljesen néma, a zöngés zárhangok középső mozzanatában pedig csak gyenge zönge hallatszik, amelyből nem lehet kideríteni, hogy a zárhangot hol képezzük. Ha a beszédfelismerő program a zárhangoknak is levágná a széleit, és csak a közepüket tartaná meg, akkor zöngétlen zárhang esetén egy néma szakaszból kellene kitalálnia, hogy a beszélő [p]-t, [t]-t vagy [k]-t mondott, vagy egyszerűen csak szünetet tartott, zöngés zárhang esetén pedig egy

olyan szakaszból kellene képzési helyre is vonatkozó információt kinyernie, amelyben ilyen információ nincs. Nem tudná tehát megkülönböztetni a két ajakkal képzett [b]-t, a fogmedernél képzett [d]-től vagy a szájpadlás közepén képzett [g]-től.

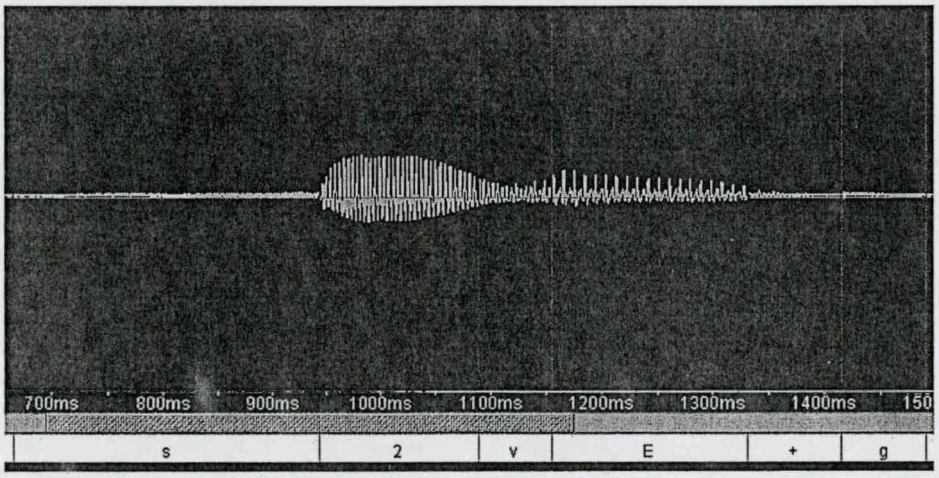
A zárhangok képzési helyét legkönnyebben a zárt követő felpattanó zörejből azonosíthatjuk, de ha ez például szó végén vagy szótag végén elmarad, akkor a záralkotásból, vagyis a megelőző magánhangzó és a zár közötti átmeneti szakaszból is kinyerhető ez az információ. Az egyetemi tankönyvek ez utóbbi lehetőséget nem említik. Papp István (1966: 86) szerint „...a kezdő és a középső mozzanat rendesen néma, tehát hangjelenség nincs jelen, a befejező mozzanat ellenben hangjelenséggel jár együtt." R. Molnár Emma és Kassai Ilona is ugyanezt tanítja: „A kiáramló hangképző levegőáram útját a szájüregben képzett zár állítja el, a beszédhang csak a zár felpattanásakor hallható pillanatnyi ideig..." (R. Molnár 1989: 43) – „A zárfeloldás nyomában keletkező hangjelenség a zárhang." (Kassai 1994: 617, 1998: 112, 2003: 531).

A záralkotás figyelmen kívül hagyása a magyarban riktán okoz gondot, mert a magyar zárhangok az esetek túlnyomó többségében felpattannak. De nem mindig. Zárhang + azonos képzési helyű orrhang találkozásakor, például *népmese, kötni, sa*[ty]*nya,* csak akkor pattan fel a zárhang zárja, ha a zárhang és a rá következő orrhang között szünetet tartunk. Ritkábban ugyan, de elmaradhat a zár felpattanása szó végén vagy egyéb zárhang + mássalhangzó kapcsolatban is. Az, hogy a magyarban is vannak fel nem pattanó zárhangok, csak a közkézen forgó egyetemi tankönyvekből maradt ki, Vértes O. András ezt többször is említi: „Bizonyos felfogás szerint a zárhangok appercipiálásakor a hallgató a legjellemzőbb mozzanatnak a hirtelen megnyitás következtésben támadó explosiós zörejt tartja. Ez a megállapítás azonban semmiképp sem általánosítható, hiszen a zárhangok egy részének zára nem nyílik fel: e zárhangok jellemző mozzanata tehát a záralkotás." (1952: 32) – „A zár felpattanása néha elmarad, például [b]+[m] kapcsolatban: *rab madár,* vagyis nem [b]-t, hanem [b¬]-t ejtünk." (1982: 158).

Ha megelégszünk egy olyan beszédfelismerő programmal, amely az esetek túlnyomó többségében jól felismeri a zárhangokat, akkor választhatjuk a könnyebbik megoldást, és figyelmen kívül hagyhatjuk a fel nem pattanó zárhangokat. Százalékosan így is nagyon jó eredményt fogunk kapni. Nem kell a folyamatos képzésű beszédhangok esetében jól bevált módszeren sem változtatnunk, ha a zárhangoknak a zárrészét különválasztjuk a felpattanástól (l. Tóth László – Kocsor András, megjelenés előtt). Ez a szegmentálási mód a záralkotást nem tekinti a zárhang részének, hanem a megelőző magánhangzó lecsengéseként elemzi. A zöngétlen zárhangok zárrészét mínusszal, a zöngésekét pedig plusszal jelöli. A beszédfelismerő program számára ezek a szakaszok csak az utánuk következő felpattanás előrejelzésére, valamint a zárhang időtartamának a megállapítására szolgálnak. Magát a zárhangot a program csak a felpattanásból azonosítja. Az első ábra a *nap* szó szegmentálását mutatja be. A hullámforma alatti átirat a SAMPA átírását követi (SAMPA 2003).

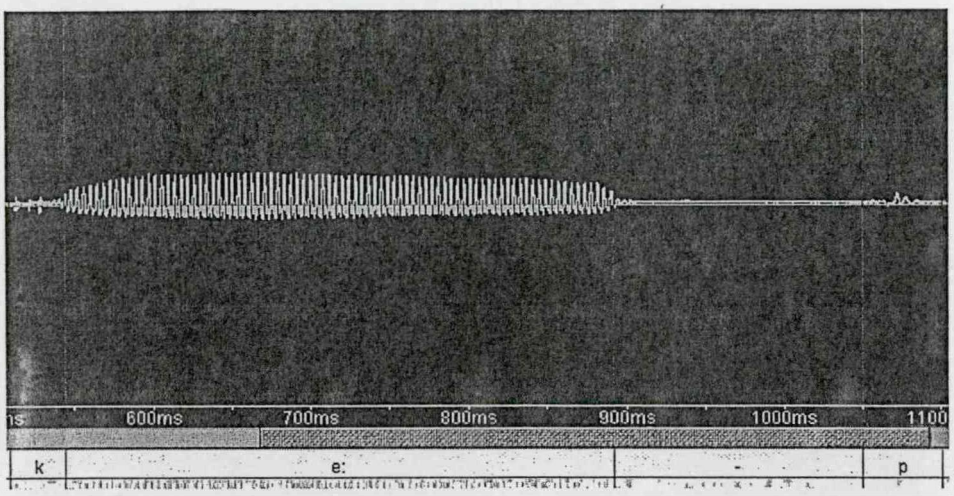


Az ábrán jól megfigyelhető a mínusszal jelölt zárszakasz utáni, amplitúdónövekedéssel együtt járó felpattanás. A szó végi zöngés zárhangok felpattanása nem ilyen erőteljes:



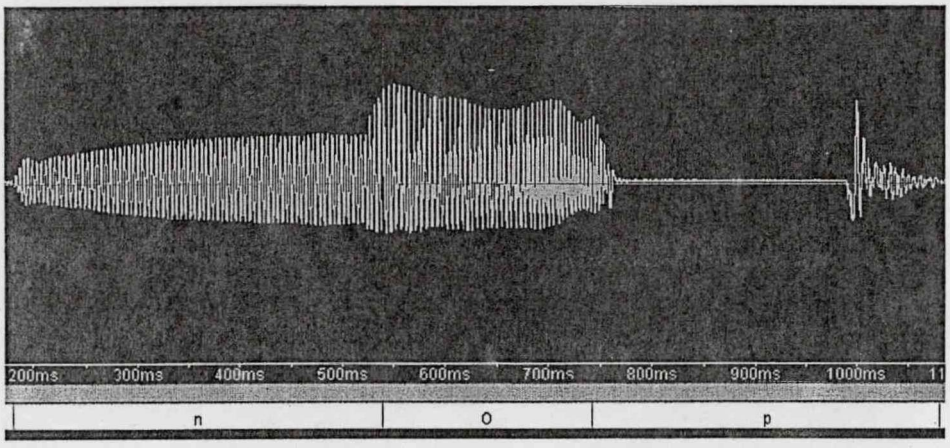
Ha lejátsszuk a *nap* [a]-ját vagy a *szöveg* [e]-jét, emberi füllel mindkét magánhangzó lecsengéséből azonosítani tudjuk az utánuk következő zárhangot, de ezt a kulcsot a beszédfelismerő program nem veszi figyelembe, mivel a magánhangzók kezdetét és végét levágja. Ezekben a példákban nincs is szükség erre az információra, hiszen a [p] és a [g] jól azonosítható a zárrész utáni, amplitúdónövekedéssel együtt járó felpattanásból.

Az alábbi ábra egy hibás szegmentálást mutat be. A fel nem pattanó [p]-vel ejtett *kép* szó hullámképében a néma szakaszt követő amplitúdónövekedés nem a [p] felpattanó zöreje, hanem valamilyen egyéb zaj:

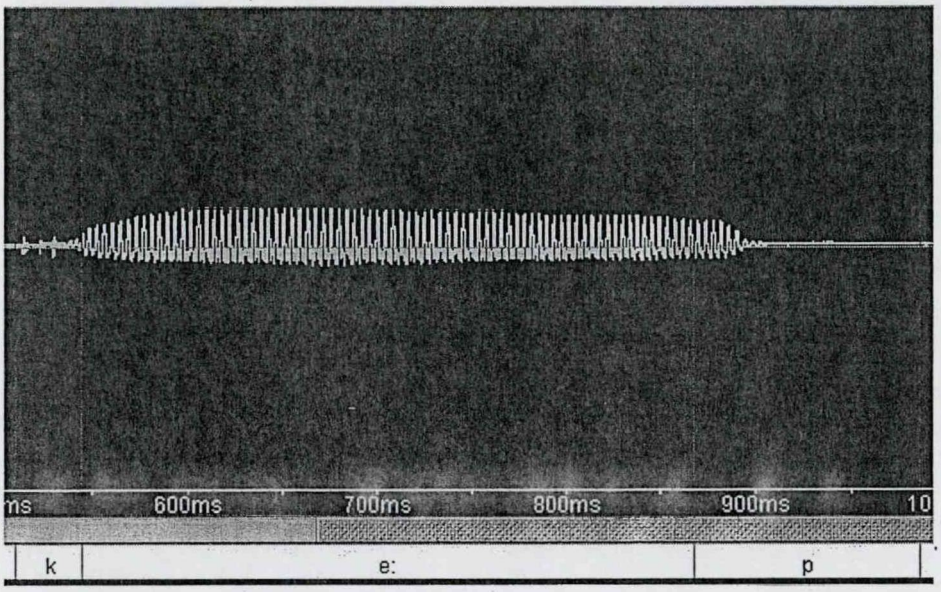


Ezt hiába húzza be a szegmentáló [p]-ként, a beszédfelismerő program nem fogja tudni belőle megtanulni a [p]-t azonosítani, mert nincs benne semmi olyan, ami a [p]-re emlékeztetne. Az anyanyelvi beszélők nem is erről ismerik fel a [p]-t, hanem az [é] végén hallható zörejről.

Ha nem ragaszkodunk ahhoz, hogy a beszédfelismerő program minden szakasznak a közepét figyelje, ahhoz viszont igen, hogy az emberekhez hasonlóan a gép is felismerje a fel nem pattanó zárhangokat, akkor a zárhangok szakaszát a záralkotástól a zár felpattanásáig kell kijelölnük:



A zárhangok felismerőmechanizmusa így valamivel bonyolultabb lesz, mert ezeknek így, a többi beszédhangtól eltérően, nem a közepét, hanem a széleit kell ellenőrizni. Ez a módszer megnehezíti ugyan a programozó munkáját, de csak az övét. A szegmentálókról nagy terhet vesz le azzal, hogy nem kell a zárhangok zárrészét a elkülöníteniük a felpattanástól. Ráadásul a felismerés hatékonyságát is javítja azzal, hogy nemcsak a felpattanó zárhangok felismerését teszi lehetővé, hanem a fel nem pattanókét is:



Irodalom

Kassai Ilona. 1994. A fonetikai háttér, in Kiefer Ferenc szerk. *Strukturális magyar nyelvtan* **2:** *Fonológia,* 581–665. Budapest: Akadémiai Kiadó.

Kassai Ilona. 1998. *Fonetika.* Budapest: Nemzeti Tankönyvkiadó.

Kassai Ilona. 2003. Fonetika, in Kiefer Ferenc – Siptár Péter szerk. *A magyar nyelv kézikönyve,* 507–548. Budapest: Akadémiai Kiadó.

Papp István. 1966. *Leíró magyar hangtan.* Budapest: Tankönyvkiadó.

R. Molnár Emma. 1989. *Leíró magyar hangtan.* Budapest: Tankönyvkiadó.

SAMPA. 2003. http://www.phon.ucl.ac.uk/home/sampa/home.htm

Tóth László – Kocsor András. Az MTBA Magyar telefonbeszéd-adatbázis kézi feldolgozásának tapasztalatai. Megjelenés alatt a *Beszédkutatás 2003* kötetben.

Vértes O. András. 1952. *Bevezetés a fonetikába.* Második, bővített kiadás. Budapest: Gyógypedagógiai Tanárképző Főiskola.

Vértes O. András. 1982. Az artikuláció akusztikus vetülete, in Bolla Kálmán szerk. *Fejezetek a magyar leíró hangtanból,* 155–164. Budapest: Akadémiai Kiadó.

Sound Transitions in Speech Recognition

György Sejtes – Gyula Zsigri

University of Szeged
sejtes@hung.u-szeged.hu, zsigri@hung.u-szeged.hu

The distinctive acoustic properties of vowels or continuant consonants are most prominent in the middle part of the arbitrarily delimited periods attributed to them. Speech recognition procedures normally ignore the edges of such units and concentrate on the middle parts. What is a good approach to vowels and continuants, does not work for stop consonants whose middle part is either silent or only has week voicing that does not give any cue about the place of articulation.

This paper compares two approaches to the problem of stop consonants. The first approach keeps the recognition procedure simple by treating each stop as if it were two segments: a pre-release segment and a release segment. The first segment is only used to tell the program that a stop release will follow soon and to measure the length of the stop. Since most Hungarian stops are released, this approach has a very good percentage of successful recognition but it puts unnecessary burden on the shoulders of the people who do the segmentation.

The other approach gives up computational simplicity and tries to recognize stops at the edges. This approach has the advantage of a simpler segmentation and adding unreleased stops to the list of recognizable speech sounds.

Eljárások idegen nyelv megértéséhez és elsajátításához

Rovny Ferenc[1,2], Páli Gábor János[3]

[1] Debreceni Egyetem Idegennyelvi Központ, 4010 Debrecen, Pf. 41.
rovnyf@flc.unideb.hu
[2] Panoráma Nyelvstúdió Kiadói Kft. 4015 Debrecen, Pf. 33.
[3] Debreceni Egyetem, Informatikai Intézet
pg0003@delfin.unideb.hu

Absztrakt. Magyarországon nagyon kevesen tudnak jól idegen nyelveket. Akik igazolható nyelvtudással rendelkeznek, azok tudásának a minősége is alacsony, különösen az idegen nyelv fonetikája terén. Igyekezünk feltárni eme súlyos hiányosság történelmi, társadalmi, politikai, nyelvoktatási hátterét és a nyelvvizsgarendszer fogyatékosságait. Foglalkozunk a fonetika-tanítás alapvető problémáival, elsősorban az angol, mint idegen nyelv kiejtése, hangsúlyozása, intonációja elsajátításának nehézségeivel. Ehhez tanácsokat is adunk. Objektív megoldásnak azonban az idegen nyelv kiejtését vizuálisan is értékelő szoftver tűnik.

1. Bevezetés

Magyarország 2004 májusában teljes jogú tagja lesz az Európai Uniónak. A sok egyéb probléma mellett Magyarország azzal is „büszkélkedhet", hogy a jelenleg csatlakozó országok közül az *idegen nyelvet tudók lakosságszámhoz viszonyított százalékos arányát tekintve az utolsó helyen áll.* Rendkívül sajnálatos, de úgy tűnik, hogy sem az ország politikai vezetői, sem a kormányzati pozícióban egymást váltogató pártok, sem az oktatást irányítók nem tették, és ma sem teszik meg a szükséges intézkedéseket a probléma kezelésére. Mintha senki sem gondolta volna végig azt, hogy miért áll Magyarország ilyen rosszul az idegennyelv-tudók számát tekintve.

2. Az idegennyelv-tudás Magyarországon: történelmi, társadalmi, politikai, nyelvoktatási háttér és a nyelvvizsgarendszer

Az ügynek ugyanis komoly történelmi és politikai háttere van. Az első kulcsszó: *Trianon.* Trianonig az ország egyértelműen *soknemzetiségű és soknyelvű* volt. Az ország jelentős területein (Felvidéken, Erdélyben, Délvidéken) *mindennapi élmény és tapasztalat* volt az „idegen nyelvvel/nyelvekkel" való találkozás. Ezeken a helyeken az egyértelműen magyar származású lakosság is tudott a magyaron kívül még egy vagy két idegen nyelven, és azt a mindennapi gyakorlatban használta is. De a német nyelvtudás elterjedtségével még a Trianon utáni Magyarországon sem volt különösebb baj, az ún. felső- és középosztály köreiben általános volt a használata.

A másik kulcsszó *1945* és az akkor bekövetkező majdnem ötven éves szovjet katonai és politikai megszállás a maga következményeivel. 1948, a fordulat éve után fokozatosan megkezdődött az idegen nyelvi tanszékek felszámolása az egyetemeken a professzorok nyugdíjba helyezésével vagy elbocsátásával. Az idegennyelv-oktatás csak az oroszoktatásra korlátozódott. Az orosz nyelv iránti idegenkedést – a magyartól való jelentős eltérésén túl – fokozta az, hogy a gyűlölt megszálló hatalom nyelve volt. Kétségtelen, hogy az '50-es évek második felében bizonyos mértékig helyreállították az idegen nyelvi tanszékeket, de a korábbi lépés egyértelműen mutatta a rendszer irányvonalát. A külföldre jutás, külföldi tanulmányutakon való részvétel (a káderek szűk felső körén kívül) elképzelhetetlen volt. Hosszú ideig tiltott, sőt büntethető is volt a külföldiekkel való bármiféle kapcsolattartás. Másrészt a '45 utáni kitelepítésekkel szinte *teljesen eltűnt a természetes másnyelvű, idegen nyelvi hangzást produkáló környezet.* Mindezek következtében kialakult egy olyan tudat – és ami még rosszabb, tudatalatti – melyben több egymást követő generáció számára *sem* volt belülről hozott természetes *érték az idegen nyelv tudása.* Kései pozitív fejlemény, hogy a '68-as gazdasági reform kapcsán beindultak a nyelvi tagozatos osztályok mind az általános iskolákban, mind a jobb gimnáziumokban.

A rendszerváltás utáni helyzet: a szigorú tankönyv és tananyag, hanganyag korlátozása ugyan már a '80-as évek lazultabb légkörében feloldódott, de a teljes szabadság csak a rendszerváltás (1989) után következett be. Ez azonban parttalanná vált, amely teljes mértékű *tananyag-zűrzavar*hoz vezetett. Az alacsony fizetések miatt a tanárok jelentős része elhagyta a pályát. Később fokozatosan *megszűnt* a szakmai követelmények bizonyos egységes, legalább közepes szintjét garantáló *szakfelügyeleti rendszer*; jelenleg gyakorlatilag nem is funkcionál. Másrészt az angol nyelvnek választott nyelvből gyakorlatilag kötelező nyelvvé válásával még a tanulók indulási motivációja is elveszett.

A felsőoktatás területén *1995*-ben következett be egy döntő lépés: ebben az évben született meg az a felsőoktatást érintő törvény, mely az egyetemi oklevél kiadását egy „C" típusú középfokú állami (ma államilag elismert) nyelvvizsga meglétéhez köti. A hiányzó *belső mély motivációt* egyszerűen egy *külső kényszerrel igyekeztek pótolni.* Azonban ez a jogszabály szakmai szempontból átgondolatlan és hibás volt, hiszen egy felsőfokú végzettségű diplomás szakembertől az lenne elvárható, hogy a saját *szakmájáról* legyen képes társalgást folytatni, vagy fordítani, sőt idegen nyelven írni. Ez azonban csak a szakmai felsőfokú nyelvvizsgának a tárgya.

A fentieken túlmenően több évtizedes gyakorlattal bíró angol nyelvtanárként rendszeresen szembeütközöm egy további tényezővel, nevezzük ezt az IR faktornak: a magyar nyelvtanulók szinte *megmagyarázhatatlan irracionális averziót* tanúsítanak az idegen nyelvek sajátosságai iránt, különösen az idegen nyelvek fonetikája, prozódiája iránt. Hallgatóim közel fele a fonetikai átírást korábban nem is ismerte, aktívan pedig csak kb. 20 %-uk használta. A végeredmény, ami egyben diagnózis is: *idegen nyelvi analfabetizmus.*

Ezért azután bármilyen *technikai eszköz vagy eljárás önmagában nem elég*, a *komplex tudati* problémákat kell helyretenni *komplex eszköztárral*, politikai, nyelvpolitikai ráhatással és propagandával, szigorú és következetes *idegennyelv-oktatási* (fonetikai és kulturális ismereteket is jelentős mértékben tartalmazó) *protokoll bevezetésével* s alaposan ellenőrzött végrehajtásával mind a közoktatási, mind a felsőoktatási nyelvtanításban.

A jelenlegi oktatási környezet és nyelvvizsgarendszer (emlékeztetőül: mely utóbbi lényegében még az elzárt, vasfüggönyös szocialista időszakban alakult ki) *nem fordít kellő figyelmet a kiejtésre, a hallás utáni megértésre* és még kevesebbet az idegen nyelvi szövegek felolvasását követelné meg. A nyelvvizsgarendszer által elfogadott átlagos értékű bizonyítvánnyal biztosan csak sikertelenül lehet részt venni egy idegen nyelvre is kiterjedő felvételi elbeszélgetésen. A hallott szöveg megértése és az idegen nyelven történő beszéd megfelelő szintje jelentős erőfeszítéseket igényel, így tehát *a kiejtéstanítás, hallás utáni megértés súlyát növelni kell mind az oktatás, mind a nyelvvizsgáztatás során.* (Jelenleg az ismert 60 %-os rendszerben egy átlagos, jó képességű középiskolás vagy egyetemista a szóbeli vizsga egyéb részein szerez annyi pontot, hogy a rosszul sikerült hallás utáni megértés eredménye nélkül is megvan a nyelvvizsgája. A kiejtés alacsony, max. 5 pontszámú értékeléséről már ne is beszéljünk!)

3. Az angol, mint idegen nyelv tanítása magyaroknak: fonetikatanítási kérdések

Nagyon sokan nem értik, hogy miért súlyos probléma az, ha az idegen nyelveket majdnem teljesen a magyar nyelv sajátosságai szerint beszélik. Nyilvánvaló, az „idegen nyelvi kommunikáció" ma Magyarországon *mesterkélt, osztálytermi körülmények* között zajlik és mintaként gyakran csak a nyelvtanár (sok esetben meglehetősen gyenge) kiejtése, hangsúlyozása és intonációja szolgál. A kommunikatív nyelvoktatás bálványának tett szolgálat következtében sokszor még kirívó esetekben sem kerül sor a kiejtés javítására. A magyar anyanyelvű tanár nyilván megszokta már a tanulók helytelen angol nyelvi prozódiáját és megérti. A Magyarországon több éve tartózkodó idegen nyelvi lektorok egy-két év alatt megtanulják, hogyan értelmezzék a magyarok által súlyosan elrontott mondatokat is. Ezáltal azonban *nem teljesülhet a nyelv*, adott esetben az idegen nyelv *funkciója* az információ és *a jelentés csorbítatlan átvitele.* A *mondat tényleges tartalma* kell, megjelenjen a nyelvtanuló kiejtésében úgy, hogy az a magyarok speciális hibáit nem ismerő anyanyelvi beszélők számára egyértelmű legyen.

Praktikus célként az alábbiakat tűzhetjük ki: 1. El kell dönteni, hogy az adott nyelvtanuló közeg számára *mi minősül követendő idegen nyelvi normának.* Az angol esetében a magyarországi nyelvtanárok képzése, a tankönyv-ellátottság és most már az EU-hoz való csatlakozás alapján is ez csak a British English lehet, annak is az idegen nyelv tanulók számára írott szótárakban, tankönyvekben szinte kizárólagosan szereplő *RP (Received Pronunciation)* [1] változata. 2. A tökéletes RP *kiejtés* elsajátítása a maga teljességében (a fonéma-egyenértékűség szintjén) az angol tanulást legkorábban is kései gyermekkorban elkezdő magyar nyelvtanulók számára megvalósíthatatlannak tűnik. Ezzel szemben a fonetikusok túlnyomó többsége a „gondozott kiejtés" elsajátítását tartja fontosnak a kiejtés területén, azaz például a *tank* és *thank* kezdőfonémájának egyértelmű elkülönítését. [2], [3] 3. *Hangsúly, szóhangsúly, mondathangsúly, beszédritmus*: ez a központi terület, amelynek az elsajátítására feltétlen törekedni kell, hiszen az adott mondat jelentése főleg a helyes hangsúlyozás révén válik egyértelművé. [4], [5] *Alapvető különbségek*: a magyar ún. *„szótagoló"* nyelv, az angol pedig ún. *„stress timing"* (hangsúly-időzítéses) nyelv. [6], [7]

Azon, hogy a magyar „szótagoló nyelv", főként az alábbiakat értjük: 1. A szabályos magyar kiejtésben minden egyes magánhangzót tisztán ki kell ejteni. 2. A szótagok között mikroszünetek vannak. 3. A tipikus *„topic-focus"* mondatok az alábbi sémát követik:

Ma_*Pé*_ter_*mo*_zi_ba_*megy*. →

A mondathangsúly a mondatkezdő *„ma"* szóra esik. 4. Minden egyes magyar szónak az első szótagja hangsúlyos. 5. A magyarok többsége tipikusan a mondat utolsó szavának hangsúlytalan szótagjait *megnyújtja, lelassítja, elhalkítja, lefelé intonálja.* Ezzel szemben egy hasonló angol mondat a következőképpen néz ki:

^

*P*eter's going to the *ci*nema to*day*.

U UU U

Az angol mondatban a mondathangsúly általában *a mondat vége felé* helyezkedik el, itt ténylegesen az utolsó szótagon van. *A magyarral ellentétben* a *„-day"* szótag a *legmagasabb, a leghangosabb és a leghosszabb.* (A szótagon *belül „emelkedő-eső"* az intonáció.)

Megoldási javaslatok a magyar nyelvtanulók számára: 1. Döntsük el, hogy hová esik a *mondathangsúlyos* szótag. (Tulajdonképpen ez fogja megadni a mondat aktuális jelentését.) 2. *Ne hangsúlyozzuk* minden szó első szótagját. 3. *Szüntessük* meg a szótagok közötti *mikroszüneteket.* 4. Koncentráljunk a (szakasz)hangsúlyos szótagok *dinamikai – hangossági* kiemelésére. Emeljük ezek magasságát a tipikus hangsúlyos magyar szótag fölé. A szakaszhangsúlyos szótagok hangosságát, intenzitását emeljük a szokásos magyar szakaszhangsúly kb. kétszeresére, a mondathangsúlyos szótagét pedig a magyarénak kb. a háromszorosára. 5. A *hangsúlytalan szótagok* időtartamát – ezzel ellentétben – jelentősen csökkentsék le *egyharmad-egynegyed* arányban. A legtökéletesebb gyógymód tulajdonképpen az angol hangsúlytalan szótagokat *mássalhangzó-torlódásként* kiejteni. Az általában szükséges [ə] úgyis automatikusan odaképződik. Egyúttal ezeket a szokásos magyar hangmagasságnál *mélyebb hangmagasságban* kell artikulálni.

A helyes mondathangsúlyozás elsajátításánál lényegesen nehezebb az *intonáció*. A magyar nyelvtanulók számára az alapvetően eltérő intonációs sémák miatt még a jobbak is sorozatos hibát vétenek. Tipikus hiba az eldöntendő kérdés helytelen intonálása:

Angol: Tune II, azaz *emelkedő* – Do you want a *drink*?
Magyar: Tune I, azaz *ereszkedő* – *Kérsz* egy italt?

Megoldási javaslatunk: a hanganyag sokszori meghallgatása, a változó dallammagasság alapos megfigyelése és utánzása. Nagy segítséget jelenthet mind a kiejtés, mind a hangsúlyozás, mind az intonáció elsajátításához Nádasdy Ádám könyve. [8]

Miután azonban szubjektíve nehéz az idegen nyelvek – mint az angol – fonetikai tulajdonságainak felismerése és megtanulása, ezt technikai eszközökkel kívánjuk segíteni. Célunk egy olyan *komplex számítógépes szoftver* kifejlesztése, mely mind a *hangsúly*, mind az *intonáció* vizuális megjelenítésére, a minta megismétlésére és annak rögzítésére, és az autentikus minta és a nyelvtanuló produktumának összevetésére, értékelésére alkalmas.

4. Számítástechnikai háttér

Pedagógiai szempontból kiemelt jelentőséggel bír a *gyakorlásra szánt beszédminták elővigyázatos kiválasztása*. Ezt jelenleg kizárólag emberi döntésre kell bízni, mivel a számítógép ebben az esetben csak támaszként alkalmazható, de ma még semmi esetre sem szabad teljes mértékben erre hagyatkozni. Emellett azonban *lehetőséget kívánunk nyújtani a gép általi döntésre is*, amennyiben a kísérletek pozitív döntési eredményekkel zárulnak. Az adekvát idegen nyelvi minták kiválasztásában viszont óriási segítséget nyújthatnak az idevágó elemzések, mivel ezek alapján a területen jártas szakemberek tapasztalataik birtokában képesek határozottan eldönteni.

Lehetőséget kell adni a programban a már meglévő *minta-adatbázis opcionális bővítésére*. A tárolást mindenképpen *meg kell előznie egy ellenőrzésnek* vagy javaslattételnek. Ezt követően ugyanúgy végre kell hajtani rajtuk a továbbiakban leírt utómunkálatokat. Fontosnak tartjuk, hogy elkülönüljenek a programmal szállított minták és a később felvettek. Ezzel segíteni kívánjuk az esetleges előre nem beszámítható hibákból fakadó kellemetlenségek előkerülését, mivel így *továbbra is kiemelve marad* a „garanciával nem rendelkező" minták halmaza.

A beszédmintákat alá kell vetni a minőség és a feldolgozási sebesség javításának szempontjából bizonyos *„cachelési" műveleteknek*, illetve normalizálni kell azokat. Cachelés alatt itt mindazon számítások előre való elvégzését értjük, amelyek egyáltalán szoba kerülhetnek a letárolandó minták kapcsán. Ezzel *már elve megtakarítunk némi időt a feldolgozási ciklus során*, amely a valós idejű kiértékelés és összehasonlítás során komoly előnynek nyilvánulhat. A normalizálás során a nyers, digitalizált mintát alakítjuk a minőségi elvárásoknak megfelelően. Ezek az elvárások is konfigurálhatóak, mivel itt is megvan a programban a tévesztés lehetősége.

A gyakorlási fázis mintegy előkészítéseként a tanuló számára megjelenítésre kerül az elismétlendő tanminta a *hangsúly és az intonáció együttes ábrázolásával*. Ezt az intonációs görbe (amelyet a *„Subharmonic to Harmonic Ratio"* kiszámolásán alapuló algoritmus [9] ad meg) megrajzolása jelenti, amit a hangsúly szerint megvastagítunk (ezt a DIN 45631 szabványban [10] ismertetettek által kapjuk meg). Ezáltal a tanuló *egyértelműen és több vetületéből* is meggyőződhet a gyakorlásra szánt mintáról. A minta megértéséhez kapcsolódóan a felhasználónak lehetősége van *gyorsítani vagy lassítani* azt, amelynek mértéke egy előre jól definiált tartományban mozoghat. Ennek az implementációs hátterét a PSOLA leírásánál [11] találjuk.

Az igény szerint a megjelenített görbéket egymásra is lehet vetíteni, így pontosabb képet lehet adni a köztük levő összefüggésekről. Ez az utóbbi művelet *automatikusan megtörténik* a kiértékelés során, hiszen számos eltérést csak így lehet hatékonyan kiemelni. A görbéket a kiértékelés során *egymásra helyezzük*, és egy előre meghatározott ún. korrektorszín segítségével bejelölődnek az eltérések. Az eltérések természetesen nem kívánnak meg hajszálpontos egyezést (hiszen erre képtelen lenne bárki!), ezért definiálni lehet egy *küszöbértéket* az eltérés mértékére. Amennyiben a kiszámított távolság meghaladja ezt, a program eltérésnek fogja minősíteni. Maguk ezek az eltérések nem jelentenek feltétlenül hibát (ennek eldöntését a nyelvtanárokra bízzuk, illetve itt is igyekszünk ebben segítséget adni a döntéshez). Hozzá kell tennünk, hogy a görbék pontos illesztése kritikus szempont, mivel ugyanúgy jól illeszkedőnek kell minősíteni egy nyújtottabb (lassabb) egy tömörebb (gyorsabb), de ívében analóg görbét is.

5. Összefoglalás

Összegzésképpen tehát megállapíthatjuk a következőket: a nyelvtanulás segítését szolgáló objektív eljárások - így mindenféle számítógépes nyelvészeti kutatás, idegen nyelvi és kétnyelvű általános és szaknyelvi adatbázisok, összehasonlító nyelvészeti kutatások, kontrasztív fonetikai vizsgálatok – jelentős támogatása szükséges és elérendő. Ahhoz, hogy az ország az Európai Unióban szerepelhessen, nemcsak szakmai tudásra, hanem ténylegesen használható nyelvtudásra is szükség van, és nemcsak a diplomások vagy annak csúcsrétege számára, hanem a középfokú korszerű szakmai képzettséggel rendelkezők számára is. A döntéshozó csúcs-elit anyagilag, például külföldi egyetemre küldéssel kompenzálja a problémát, a társadalom szélesebb rétegei pedig nem is tudnak róla.

Ez egy *komplex*, társadalmi-tudati-törvénykezési-oktatáspolitikai-nyelvvizsgáztatási *probléma*, melyben nyilvánvalóan a tudati és egyéb körülmények megváltoztatása a döntő. A téma mindenki számára nyilvánvaló feltárására azonban *technikai megoldás* szükséges, hogy a tömeges hibás gyakorlat vizuálisan is érzékelhetővé váljon, másrészt az objektív gyakorlásra is lehetőség nyíljon. Ehhez kívánunk segítséget adni kifejlesztés alatt álló speciális szoftverrendszerünkkel.

6. Hivatkozások

1. Gramley, S., Patzold, K-M.: A Survey of Modern English. Routledge, London New York (1992) 304-314
2. Várnagy, E.: Adalékok a magyar és angol fonémarendszer kontrasztiv vizsgálatához. In: Horváth, M., Temesi, M., (szerk..) Összevető nyelvvizsgálat, nyelvoktatás.Tankönyvkiadó (1972) 264-270
3. Horváth, T.: A külső és belső hangkontrasztivitás szerepe az idegen nyelvek oktatásában.In: Horváth, M., Temesi, M., (szerk..) Összevető nyelvvizsgálat, nyelvoktatás.Tankönyvkiadó (1972) 282-287s
4. Varga, L.: A Contrastive Analysis of English and Hungarian Word Stress. In: Dezső, L., Nemser, W. (eds. szerk.) Studies in English and Hungarian Contrastive Linguistics. Akadémiai Kiadó (1980) 233-244
5. Allen, W. S.: Living English Speech. Longmans, Green & Co. Ltd. (1969) 1-32
6. Roach, P.: English Phonetics and Phonology. Cambridge University Press (1985) 102-104
7. Gramley, S., Patzold, K-M.: A Survey of Modern English. Routledge, London New York (1992) 109-110, 111-113
8. Nádasdy, Á.: Angol kiejtési gyakorlatok. Tankönyvkiadó (1987)
9. Sun, X. J.: A Pitch Determination Algorithm Based on Subharmonic-to-Harmonic Ratio. Department of Communication Sciences and Disorders, Northwestern University (2001)
10. Zwicker, E., Fastl, H.: Psychoacoustics: Facts and Models – Second Updated Edition, Springer-Verlag, Berlin Heidelberg New York (1990, 1999)
11. Götzen, A. D., Bernardini, N., Arfib, D.: Traditional Implementations of Phase-Vocoder: The Tricks of the Trade, Proceedings of the COST G-6 Conference on Digital Audio Effects (DAFX-00), Vienna, Italy (2000)

Procedures for the Comprehension and Acquisition of Foreign Languages

Ferenc ROVNY, *UD Foreign Language Centre & Panorama LS Publishing;*
Gabor Janos PALI, *UD CSc. student; rovnyf@flc.unideb.hu, pg0003@delfin.unideb.hu*

Keywords: *CALL, Foreign Language Learning, Listening Comprehension, Contrastive-Comparative Linguistics, Pronunciation Standard, Stress, Intonation; Speech Processing algorithms: PSOLA, SHRPD, DIN45631*

Out of the computational efforts to help foreign language learning especially the results hitherto achieved in the field of Listening Comprehension (LC) and Speech Acquisition have not treated the issue in its complexity. E.g. the English language significantly differs from the Hungarian in its phoneme set, stress and intonation. In English, words go through marked quantitative and qualitative transitions owing to speech rhythm. The main difference between the two languages is that the Hungarian has a *„syllabic"* character while the English has a *„stress-timing"* one. That is why LC and correct English speech constitute serious trouble for Hungarian learners of English. *The problem cannot be treated in a proper objective way within the usual framework of language learning and teaching.* So we have come to the conclusion that we should try to develop a special kind of software for the enhancement of the acquisition of foreign languages (especially that of the English). The software is being developed in *MatLab* environment at present and for the sake of user-friendliness a Graphical User Interface is also added to it.

The major components and steps of the task are as follows: 1. It is to be defined what can be considered an *adequate foreign language pattern.* 2. It is also to be decided that the software should handle *stored samples only* and/or *any optional authentic foreign language speech samples.* 3. The next step is the *digitalisation of speech patterns* and carrying out the necessary filtering and post-processing. 4. The *stress and intonation curves* belonging to the stored sentence patterns are to be prepared and drawn in an unambiguous form. This way the learner can have a *visual image* of how the given sentence is to be produced according to the rules of English speech. 5. The learner will be able to *slow down* (or even *speed up* if he/she wishes so) the heard speech pattern in a pre-defined range and that can help him/her in the comprehension of the pattern. 6. *Practising:* the learner can reproduce the patterns, meanwhile the reproduced patterns are being stored. The stressed parts of both the heard and reproduced speech patterns are *visually marked,* at the same time the intonation curve and the speed also appear in the function of time. 7. *Evaluation and marking:* the authentic pattern and its reproduction by the learner are *visually compared.* (At present, phonemic level comparison is not planned. We concentrate on stress and intonation as they are the two main pillars of LC and the differentiation of meaning.)

The optional slowing down of samples is implemented by the ***PSOLA*** *(Pitch Synchronous Overlap and Add)* algorithm. The algorithm requires pitch detection, i. e. the estimation of fundamental frequency. The method chosen to do this is the ***SHR PD*** *(Subharmonic to Harmonic Ratio Pitch Detection)* algorithm. The *distortionless slowing down* of the sample is also important. The ***DIN45631/ISO532B*** standard defines the stress function. The intonation function given by the PDA must be combined with the computation of *short-time energy* or *loudness.* The comparison is implemented by the method of *computation of Mahalanobis distance.* The result of the computation serves as the basis of the evaluation.

Nyelvészeti és számítástechnikai módszerek az igazságügyi nyelvészetben

Hunyadi László, Abari Kálmán, Tóth Enikő

Debreceni Egyetem

Általános és Alkalmazott Nyelvészeti Tanszék

4010, Debrecen, Pf. 24.

hunyadi@llab2.arts.klte.hu
abarik@pmail.arts.klte.hu
teniko@pmail.arts.klte.hu

Absztrakt. Cikkünk az igazságügyi nyelvészet ágazatába sorolható esettanulmány ismertetése, melynek során nyelvészeti és számítástechnikai eszközöket is alkalmazva arra a kérdésre kerestük a választ, vajon igazolható-e az, hogy egy adott digitális módon készült hangfelvételt digitálisan manipuláltak. Az elemzés célja annak kiderítése volt, hogy a hangfelvételen található-e vágásra, megszakításra utaló akusztikai jel. Interdiszciplináris elemzést végeztünk, három, egymástól jól elkülöníthető, szemantikai, kísérleti fonetikai, illetve számítástechnikai szempontból vizsgáltuk a fenti kérdést. Ezen módszerek együttes alkalmazása alapvetően sikeresnek bizonyult az eredeti kérdés megválaszolásában. A javasolt új módszerek a nyelvészet és a számítástudomány egyéb területein is hasznosak lehetnek.

1. Bevezetés

Az igazságügyi nyelvészet az alkalmazott nyelvészet ágazatai közé tartozik, eredményei elsősorban bírósági tárgyalásokon alkalmazhatók. Viszonylag rövid története ellenére egyre nagyobb jelentőséggel bír, elsősorban az új számítástechnikai technológiák megjelenésének köszönhetően. Két szempontból is figyelemreméltó a hangfelvételek készítésének lehetősége. Először, amikor technikai szempontból megvalósíthatóvá vált a hangfelvételek készítése, az igazságügyi nyelvészetben is megjelent a beszélő hangfelvétel alapján való azonosításának problémája. Másodszor, a hangfelvételek készítésének lehetősége együtt jár azok esetleges manipulálásával is. Míg a hagyományos hangszalagokkal való manipulálást viszonylag egyszerű kimutatni (a szalag fizikai károsodása vagy a szalagon lévő, manipulálásra utaló elektronikus jegyek révén), a digitális hangfelvételek esetén ugyanez a feladat igazi kihívást jelent. Mivel a digitális hangfelvételek számjegyek sorozataiként realizálódnak, feltehetjük, hogy a hangfelvétel manipulálása egyszerűen a számjegyek sorozatainak módosítását jelenti. A digitális hangszerkesztőkkel ez meg is valósítható,

viszont nyitva marad az a kérdés, vajon kimutatható-e a hangfájlok effajta módosítása.

A következőkben egy valós eseten alapuló vizsgálatot mutatunk be, amelynek célja annak megállapítása volt, hogy a rendelkezésünkre bocsátott digitális hanganyagot manipulálták-e.

2. A probléma

Az alábbiakban bemutatott vizsgálat nyelvészeti és számítástechnikai eszközök alkalmazásával válaszolja meg a következő kérdést: igazolható-e, hogy egy digitális módon készült hangfelvételt digitális módon manipuláltak. A kérdésfelvetés és annak vizsgálata valós eseten alapul.[1]

A hatóságoktól egy audio CD-t kaptunk, amely a kérdéses értekezlet jegyzőkönyvét tartalmazta wav formátumban. Továbbá rendelkezésünkre bocsátották azt a merevlemezt, amelyre a felvételt eredetileg rögzítették, azonban információik szerint a hangállományt korábban letörölték. Így ki kellett dolgoznunk egy olyan eljárást, amely segítségével megállapítható, hogy a merevlemezről letörölt fájlok egyes részei valamilyen mértékben helyreállíthatók-e.

Ennek megfelelően először megvizsgáltuk, hogy a CD-n lévő hanganyagon találunk-e manipulálásra utaló jeleket, másodszor megpróbáltuk a letörölt hangfájlokat, a lehetőség szerint, azonosítani. A hanganyagot először nyelvészeti elemzésnek vetettük alá, majd figyelembe vettük, hogy digitális formában lévő anyagot elemzünk, ezért elektronikus reprezentációjuk formáját is vizsgáltuk.

3. Módszerek

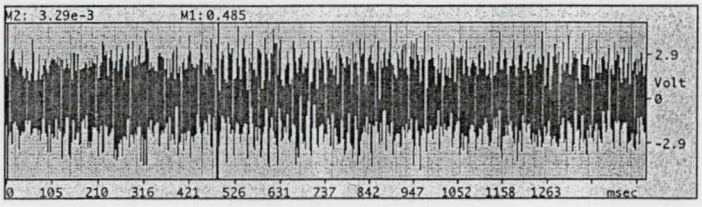
3.1 A lehetséges manipulálására utaló digitális jelek azonosítása

Az igazságügyi nyelvészetben korábban tudomásunk szerint csak analóg felvételek manipulálásával foglalkoztak (ld. Gruber et al, 1993, Gruber – Posa, 1995). Esetünkben azonban a digitális adatreprezentálás természete miatt a már kidolgozott módszert nem alkalmazhattuk. Így először egy előzetes kísérletet végeztünk annak meghatározására, hogy a szándékos manipulálás vajon hagy-e azonosítható jelet az anyagban.

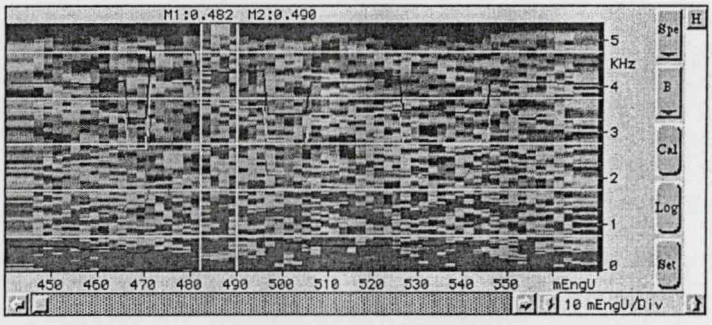
---

[1] Egy kft. elnökségi értekezletéről kétféle jegyzőkönyvet nyújtottak be. Az egyik jegyzőkönyvet a már régóta hatályban lévő ügyvezető igazgató írta alá, a másik jegyzőkönyvet pedig az ezen az ülésen állítólagosan megválasztott igazgató. A korábbi igazgató az értekezletről digitális hangfelvételt készített. Az újonnan megválasztott igazgató viszont amellett érvelt, hogy a hangfelvétel nem hiteles, valószínűleg manipulálták, és a releváns részeket, amelyek az általa benyújtott jegyzőkönyvben találhatók meg, kivágták. Feladatunk tehát az volt, hogy eldöntsük, történt-e manipulálás, melyik jegyzőkönyv hiteles.

Az elemzést a hanghullám spektrografikus reprezentációjára alapoztuk, azt feltételezve, hogy a manipulálás szóhatároknál a legvalószínűbb, különösen a környezeti zajt, de beszédjelet nem tartalmazó szakaszokban (továbbiakban zajszegmensekben) jellemző. Ennek modellezésére eltávolítottuk egy adott zajszegmens egy részét, majd a hátramaradt két részt konkatenáltuk. A szokásos felbontást nagymértékben növelve a spektrogramon láthatóvá vált a konkatenálás helye, egy digitális jel, amely a tekintett hangfájl minden más szegmensétől egyértelműen különbözött. Az 1. ábrán látható a konkatenálás hanghulláma, míg a 2. ábra ugyanazon hanghullám nagy felbontású spektrális képe.



1. ábra Az 1., 3. zajszegmens összefűzése 0,484 ms-nál, miután a 2. zajszegmenst eltávolítottuk közülük



2. ábra A 1. ábrán látható összefűzött hanghullám spektrogramja 0,485 ms körül nagyítva

Ahogy a 2. ábrán látható, a digitális manipulálás helyén egyértelmű nyom látható, amely legalább három szignifikáns jellemzővel bír:

(1) a manipulálás helyén a jelek szabályos függőleges eloszlást mutatnak, vagyis minden frekvenciánál megnövekedett az intenzitás

(2) ez a növekedés szabályos horizontális (idő szerinti) eloszlást is mutat, szimmetrikus struktúra jelenik meg a kb. 0,008 ms szélességű ablakban: az intenzívebb centrális eloszlás mindkét oldalán egy-egy kevésbé intenzív 0,002 ms-os szegmens helyezkedik el

(3) ezen intenzitás-megoszlás maximális értéke jóval meghaladja a felvett szokásos zaj mért maximumát (5 kHz)

Következő feladatunk az volt, hogy megmutassuk, hogy a kísérlet során talált digitális jel (ld. 2. ábra) megbízható eszközt jelent a hangfelvétel digitális

manipulálásának kimutatására. A bizonyítást az alábbi négy lépésben végeztük: igazolni kívántuk, hogy: (1) az elemzendő anyag zajszegmenseinek vágása hasonló nyomot hagy hátra, (2) beltéri körülmények között rögzített zaj tetszőleges szegmensének eltávolítása hasonló nyomot hagy hátra, (3) a vágás nem szoftverfüggő, (4) a vágás nem hardverfüggő.

Ennek érdekében az adott zajból és más zajt tartalmazó felvételekből statisztikailag megfelelő számú vágást hajtottunk végre, majd ezeket megismételtük különböző szoftver és hardver platformokat használva.[2] A tapasztalatokat összegezve megállapítottuk, hogy a feltevésünk helytálló.

3. 2 Szemantikai elemzés

A hanganyagot szemantikai elemzésnek is alávetettük, mely a rendelkezésre álló szöveg tartalmi integritását vizsgálta, abból a feltevésből kiindulva, hogy a csonkítás általában a tartalmi összefüggések sérülésével jár. A szemantikai elemzés során a szövegben olyan részeket kerestünk, amelyek szemantikailag inkoherensek, vagyis vágásra utalhatnak, majd ellenőriztük, vajon tartalmaznak-e ezek a részek manipulálásra utaló digitális jeleket.

A szemantikai elemzés célja tehát annak kiderítése volt, hogy a beszélgetés struktúrájában található-e valamilyen információs hézag, hiányzó összefüggés, így a szöveg strukturális koherenciáját vizsgáltuk (ld. Brown – Yule, 1993). A szöveget abból a szempontból tanulmányoztuk, hogy megállapíthassuk, vajon sérült-e a szövegkohézió, illetve koherencia, vagyis a szöveg kohéziós összefüggéseit vizsgáltuk: a névmások használatát és antecedensükkel való anaforikus kapcsolatukat és a strukturális viszonyokat (pl. elliptikus szerkezetek, szintaktikai ismétlés, igeidők használata és ezek sorozatainak konzekvenciája). Szükségesnek tartottuk a beszélgetés információ-struktúrájának, a beszélők által közvetíteni szándékozott feltételezett üzenetek elemzését is. Azt a vizsgálati szempontot tartottuk szem előtt, hogy tartalmaz-e az anyag olyan hézagokat, amelyek nem rekonstruálhatók a háttér-információ alapján (vagyis a szöveg lehetséges manipulálására utaló jeleket kerestünk).

3.3 A merevlemezen lévő fájlok manipulálásának statisztikai azonosítása

Elemzésre ugyan megkaptuk azt a merevlemezt, amely feltételezhetően az eredeti hangfelvételt tartalmazta, azonban azt később letörölték, így nem állt módunkban a kérdéses értekezlet átiratát egybevetni az eredeti felvétellel. Feltételeztük, hogy az eredeti felvétel néhány töredéke még megtalálható a merevlemezen, annak ellenére, hogy hagyományosan ezeket a fájlokat nem tekintik rekonstruálhatónak. Ezen hangfájlok/hangfájltöredékek azonosítására a nullátmenet (ZC) kiszámítását választottuk, az alábbi megfontolások alapján.

---

2 Kísérleteink nagy részét Macintosh számítógépen, SoundScope/16 felhasználásával végeztük, de egy Intel-alapú PC-n Cool Edit programmal végzett kísérletek ugyanezt az eredményt hozták

A beszéd legfontosabb jellemzője, hogy zöngés, zöngétlen és csendes szegmensek sorozataiból áll. Általában véve a zöngétlen hangoknak magasabb a frekvenciája (így ZC értékük is magasabb), mint a zöngés hangoké, míg a csend oszcillációja (és ZC értéke) gyakorlatilag 0.

Húsz különböző fájltípust vizsgáltunk, azokat, amelyek a legnagyobb valószínűséggel fordulnak elő egy átlagos számítógépen (ld. az 1. táblázatot)[3], és minden fájltípusra húsz fájlt tekintettünk, egyenként 10 Mb körüli mérettel.

1. táblázat Nullátmenetek számának átalaga és szórása

	Átlag	Szórás
avi	123.15	122.84
bmp	96.60	155.21
chm	480.90	12.51
dat	157.30	210.22
dll	333.35	97.11
doc	169.90	110.63
exe	370.20	87.91
gif	510.40	1.50
hlp	277.90	49.05
html	0.70	3.05
hpg	501.85	7.47
mp3	481.20	59.91
pdf	310.70	122.35
rtf	7.30	30.47
sys	381.05	87.30
txt	3.45	10.52
txt2	151.00	19.10
wav	98.10	18.54
wav2	73.20	11.85
xls	197.60	63.04
zip	496.75	11.03

Az összesítő táblázat első két oszlopa az egyes fájltípusok 1 Kb-jára eső nullátmenetek számának átlagát és azok szórását tartalmazza. A hangfájlokra (wav és wav2) ezek az értékek 98,1 és 73,2, a szórás viszonylag kicsi (18,54, valamint 11,85). Az adatok elemzésekor jelentős eltérést találtunk a chm, dll, exe, gif, hlp, jpg, mp3, pdf, sys, txt2 és zip fájltípusok valamint a html, rtf és txt fájlok között, az előbbiek magasabb, az utóbbiak alacsonyabb értékekkel rendelkeznek. Az avi, bmp, dat, doc és xls fájlokhoz tartozó átlagértékek közel voltak a hangfájlok értékeihez, de szórásaikban jelentősen eltértek azoktól.

A fenti kísérleti adatok azt mutatják, hogy egy sztenderd adatrekonstrukciós szempontból nézve reménytelennek tűnő esetben a bitszintű töredékek statisztikai elemzése bizonyos lehetőséget nyújthat az adatok helyreállítására.[4]

3 A fájltípusok között a txt kiterjesztés az ékezet nélküli ASCII kódú fájlokat reprezentálja, míg a txt2 kiterjesztés az ékezeteseket. A wav és a wav2 kiterjesztés 22,050 Hz-es 16 bites magyar beszélt adat reprezentálása, de a wav2 nem tartalmaz zajokat.

4 Következtetések

Az előző részben ismertetett módszereket alkalmaztuk annak érdekében, hogy választ kapjunk az eredeti kérdésre, vajon manipulálták-e a digitális hangfájlt. Összesítve az adott esetre vonatkozó vizsgálati eredményeket arra a következtetésre jutottunk, hogy (1) a hangfelvételben nem találtunk digitális manipulálásra utaló nyomot és (2) feltéve, hogy az általunk azonosított nyom az egyetlen minta, amellyel a digitális manipulálás azonosítható, a fájlon digitális manipulálást nem hajtottak végre.

Cikkünkben egy adott igazságügyi eset kapcsán, nyelvészeti és számítástechnikai eszközöket is alkalmazva arra a kérdésre kerestük a választ, vajon igazolható-e az, hogy egy adott digitális módon készült hangfelvételt digitálisan manipuláltak. Megmutattuk, hogy a hagyományos spektrografikus elemzés során a beszédhang új dimenziókra, szinte mikroszkopikus szegmensekre való szűkítésével egy olyan tartomány tárható fel, amelyben a hangfájl digitális manipulálásának nyomai kimutathatók. Ezt az eljárást hagyományos szemantikai tartalmi elemzéssel kombináltuk, amely során az egyik megközelítés kimenete bemenetként szolgálhatott egy másik eljáráshoz. A hangfájlokra alkalmazott digitális manipulálás jellemző nyomainak azonosítása mellett egy statisztikai eljárást is kidolgoztunk a különböző típusú adatfájlok azonosítására, amivel elősegíthetjük a merevlemezeken lévő már letörölt, header nélküli fájltöredékek azonosítását.

Irodalom

1. Borbándi J., Csáki, E., Nemes L. (szerk): A nyelvész szerepe a kriminalisztikában: 7. Országos Kriminalisztikai Tanácskozás. BM, Budapest, 1977
2. Brown, G., Yule, G.: *Discourse analysis. (Diskurzuselemzés)* Cambridge University Press, Cambridge, 1993
3. Gruber, J. S., Posa F. T.: Voicegram identification evidence. ('Spektrogramok' azonosítása) *American Jurisprudence*, 54., 1995
4. Gruber, J. S., Posa F. T., Pellicano A. J.: Audiotape recordings: evidence, experts and technology. (Magnófelvételek: bizonyíték, szakértők, és technológia) *American Jurisprudence*, 48., 1993
5. Hunyadi, L., Abari, K., Tóth, E.: *Forensic Linguistics: its Contribution to Humanities Computing.* Literary and Linguistic Computing, Vol. 18, No. 1, 2003
6. Kontra, M.: Nyelv és jog. In: Kiefer, F. (szerk) A magyar nyelv kézikönyve. Akadémiai Kiadó, Budapest, 2003
7. Schanze, H. A. European perspective for the PC age. (A PC korszak európai perspektívái) *Literary and linguistic computing*, 5(2), 1980: 171-3

[4] A történethez hozzátartozik, hogy annak ellenére, hogy a merevlemezen lévő fájlok manipulálását kimutató statisztikai eljárás jól működött kísérleti körülmények között, a kidolgozott eljárást nem alkalmazhattuk a kérdéses esetben, ugyanis a merevlemezt, amely vélhetőleg az eredeti hangfelvételt tartalmazta, újraformázták. Ennek következtében nem állt rendelkezésünkre adat, amin a statisztikai elemzést elvégezhettük volna.

Linguistic and Computational Methods in Forensic Linguistics

László Hunyadi, Kálmán Abari, Enikő Tóth

University of Debrecen
Department of General and Applied Linguistics
4010, Debrecen, Pf. 24.

hunyadi@llab2.arts.klte.hu
abarik@pmail.arts.klte.hu
teniko@pmail.arts.klte.hu

Keywords: forensic linguistics, experimental phonetics, digital hanganyag, statistical methods, computational soundanalysis, semantic analysis

The paper is a report on a case in forensic linguistics in which linguistic and computational approaches are combined to answer the question whether it can be proved if a digital recording has been tampered with. With the growing use of digital applications the chances of digital forgery are significantly increasing, accordingly, the detection of tampering with audio recordings is also becoming an important task for forensic linguists. In the given case, we assumed that the most straightforward way of tampering with the given digital audio recording might have been the removal of some material and so our aim was to identify the location of this kind of tampering in the file. Due to the complexity of the given task the approach presented is interdisciplinary: first, it uses a traditional semantic analysis to identify possible discontinuous segments of the recorded text, second, it introduces an experimental phonetic approach to identify cues of the digital cutting of the audio signal, third, it applies statistical calculations to specify the bit-level characteristics of audio recordings. The combination of these measurements proved to be quite helpful in answering the initial question, and the proposed new methodologies can be used in further areas of linguistics and computation.

Beszélő fej

Czap László[1], Mátyás János[2]

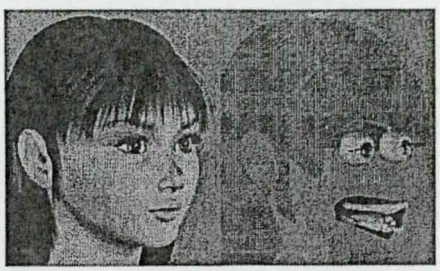
[1] Miskolci Egyetem, Villamosmérnöki Intézet, Automatizálási Tanszék
3515 Miskolc, Egyetemváros
czap@mazsola.iit.uni-miskolc.hu
[2] Észak-Magyarországi Regionális Munkaerőfejlesztési és Átképző Központ
3518 Miskolc, Erenyő u. 1.
matyasj@mail.erak.hu

Abstract. Magyar nyelvű, vizuális szövegfelolvasó fejlesztéséről számolunk be cikkünkben. Az animáció háromdimenziós fejmodell mozgatásán alapul. Az artikuláció kialakításához felhasználtuk a fellelhető hangalbumok anyagát, a dinamikus vizsgálatnál saját vizuális beszédfelismerési kutatási eredményekre támaszkodtunk. A koartikulációs hatások figyelembe vételéhez a jellemzőket domináns, rugalmas és határozatlan osztályokba soroltuk, ezek alapján határoztuk meg a mozgásfázisok közötti interpolációt. A természetesség javítása érdekében többek között álvéletlen fejmozgásokat és pislogást programozunk. A szemöldök mozgatása fontos szerepet játszik a gesztus kialakításában. A fejmodell működtetése során megvalósítjuk alapérzelmek kifejezését is. A cikk végén kijelöljük a továbbfejlesztés irányait.

1 Bevezetés

Mindenki előtt ismert, hogy a beszéd érthetőségét javítja, ha látjuk a beszélő személy arcát, ezzel együtt az artikulációját. Ez a vizuális információ különösen sokat segít zajos környezetben és hallássérültek esetében. A gépi beszédkeltés jól kidolgozott rendszereinek természetes kiegészítője a mesterséges beszélő fej. Az arcanimáció megvalósítása a beszédartikuláció modellezésére mindössze két évtizeddel ezelőtt kezdődött. A mai szemmel kezdetleges eszközökkel végzett első próbálkozások a vizuális beszédszintézis kezdetét jelentették. A 3D modellezés fejlődése, a számítástechnikai eszközök kapacitásának robbanásszerű bővülése és a természetes artikuláció analízise életszerű, fotorealisztikus finomságú modellek kidolgozását tette lehetővé.

Az elmúlt évtizedben a terület dinamikusan fejlődött, egyre több alkalmazás jelenik meg. Az ember-gép kapcsolatban új távlatokat nyithat az audio-vizuális beszédszintézis és beszédfelismerés. Dialógus és oktató rendszerekben az érthetőséget és az attraktivitást nagyban javítja a beszédanimáció. Multimédia alkalmazásokban a virtuális bemondó vagy szereplő tágítja a művészi szabadság határait. Hallássérültek beszélni tanítását segítheti a helyesen artikuláló virtuális bemondó, amely átlátszó arcával a természetes beszélőnél jobban megmutatja a hangképzés részleteit.



1. ábra Fotorealisztikus és transzparens megjelenítés

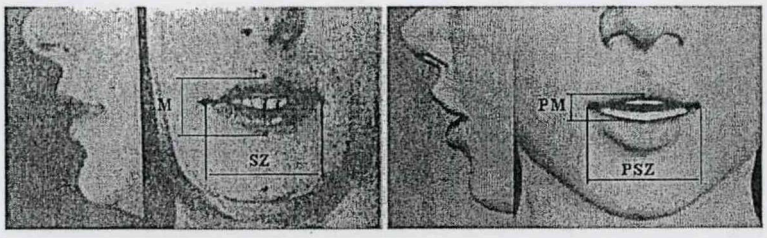
Hangvezérelt beszélő fejek fejlesztésén dolgoznak hallássérültek segítésére távközlési alkalmazásokban. A fejlett magyar nyelvű akusztikus beszédszintézis mellett hiánypótló célzattal kezdtünk vizuális beszédszintetizátor fejlesztéséhez.

2 A beszédanimáció

Az első működőképes vizuális beszédszintetizátorok kétdimenziós modell mozgásfázisainak előállítására épültek, kezdetben előre tárolt képek előhívásával. A kulcskeretek közötti fázisokat gyakran morfológiai módszerekkel állították elő. A kétdimenziós modell nem teszi lehetővé a természetes fejmozgások, a beszédet kísérő gesztusok és érzelmek kifejezését. A testmodellezés fejlődése a háromdimenziós modellezésre terelte a kutatók figyelmét. A 3D modellek egyik típusa az arcizmok megfeszítésével szimulálja az arckifejezéseket. Az ilyen modellek valósághű eredményt nyújtanak, de a kívánt arckifejezés előállítása rendkívül számításigényes és a valóságos izomtónusok nem mérhetők. Ma még ígéretesebb a pusztán felületi hatásokat utánzó, a bőrszövettel borított drótváz alakítására alapozott animáció. Ennek paraméterei megfigyeléssel, vagy képfeldolgozási módszerekkel természetes beszélők képeiről leolvashatók. [1] Minden modell mozgatásánál külön figyelmet kell fordítani a jellemzők összehangolt változtatására, mert könnyen természetellenes hatás alakulhat ki. Például az alsó fogsor és az áll független mozgása groteszk hatást kelt.

2.1 A beszéd vizuális alapegysége

A beszéd legkisebb akusztikus egységének, a fonémának vizuális megfelelője, a *vizéma*. A vizémák készlete szűkebb a fonémákénál, hiszen néhány fonéma artikulációja vizuálisan megegyezik. Nem látható pl. a zöngésség, de a képzés helyében megegyező, időtartamban vagy intenzitásban eltérő hangok is azonos artikulációs mozgásokkal jelennek meg. A hangképző szervek jellemző helyzete magyar beszédhangokra megtalálható alapvető munkákban [3], [4], [5]. A 2. ábrán példát mutatunk be arra, hogy mennyire hasonló egy hagyományos labiogram [4] és egy 3D-s beszélő fejen beállított ugyanazon hangra jellemző artikuláció.



2. ábra A minta fotolabiogram és a renderelt 3D fejmodell

A magyar beszédhangok vizéma készletét a [4]-ben megadott mintaszavak artikulációs jellemzőiből alakítottuk ki. Az eredményt az 1. táblázat mutatja, a hangokat a magyar helyesírási betűképükkel jelöljük.)

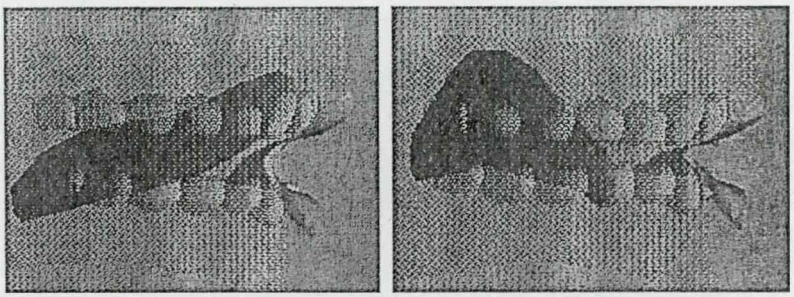
1. táblázat A magyar nyelv vizéma készlete

Magánhangzók	Mássalhangzók
e	b, p, m
é	f, v
I	t, d, n
ö, o	r
ü, u	sz, z, c, dz
á	l
a	s, zs, cs, dzs
	ty, gy, j, ny
	k, g
	h

Néhány megjegyzés a vizémák osztályozásához:

- a csoportosítás elsősorban ajakforma alapján történt, a nem látható nyelvállás eltérő lehet (pl.: o-ö, u-ü)
- a nem jelzett hosszú magánhangzók a rövid párjuknál szűkebb szájnyílással vannak jelen
- az artikuláció előállításához ennél bővebb készlettel dolgozunk

Az eddig megjelent beszédhangok atlasza [3], illetve magyar hangalbumok [4, 5] alapján meghatározhatók a vizémák legfontosabb paraméterei, ezekből alakul ki az a kulcskeret (keyframe) készlet, amely az artikuláció kiindulási alapja [6]. A legfontosabb jellemzők az ajkak és a nyelv működtetéséhez tartoznak. Az alapvető ajak jellemzők: nyitás (tág-szűk), szélesség (széles-keskeny), Az ajkak nyitása szoros összefüggésben van az állkapocs mozgásával (nyitott - zárt). A száj szélessége tehát az ajaknyitással és az ajakkerekítéssel, illetve az ajakréssel, áll összefüggésben. Az állkapocs helyzete a nyitás mellett a fogak láthatóságával is összefügg. A nyelvállást (2. ábra) a nyelv függőleges helyzete (fent-lent), vízszintes mozgása (elül-hátul), hajlítása (domború-homorú), és a nyelvhegy formája (széles-keskeny, vékony-vastag) befolyásolják.



3. ábra Jellemző nyelvállások: baloldalon az n-re, jobbra a k-g hangokra

A statikus jellemzők alapján beállíthatók a beszédhangok állandósult szakaszára jellemző artikulációs paraméterek, kulcskeretek.

2.2 Dinamikus működés

A folyamatos magyar beszéd dinamikus jellemzőinek átfogó leírása még várat magára. Az analízis során a hangalbumokban található pillanatképek korlátozottan használhatók, és csak a mintaszavakra vonatkoztathatók. A dinamikus analízis másik forrása a saját, vizuális beszédfelismerési kutatásaink során nyert eredményekből összeállított adatbázis [7]. Ebből származnak az ajkak nyitásának és szélességének időbeli változására vonatkozó adatok, valamint a nyelv és a fogak láthatóságát reprezentáló intenzitás faktor, a szájüregre vonatkozóan. Ezek a kulcskeretek közötti interpoláció megválasztásában nyújtanak segítséget.

A koartikulációs hatások figyelembe vételéhez túl kellett lépnünk az úgynevezett „keyframe" modellen. A vizémák minden jellemzőjét (például ajak- és nyelvállások) osztályoztuk dominans jellegük alapján. Egyes paraméterek a környezettől függetlenül felveszik jellegzetes értékeiket, mások a környezetükbe simulnak. A vizuális beszédfelismerés adatainak szórása alapján a vizémák jellemzőit három kategóriába soroltuk:

- *domináns* – nem enged koartikulációs hatásoknak
- *határozatlan* – a környezete alakítja ki az adott jellemzőt
- *rugalmas* – a környezete befolyásolja az adott jellemzőt

Példaképpen megadjuk a vizémák ajakformára és a nyelv vízszintes helyzetére vonatkozó csoportosítását (2. táblázat):

2. táblázat Dominancia jellemzők az ajakformára nézve

Domináns	magánhangzók, s, zs, cs, dzs
Határozatlan	t, d, n, r, l, ty, gy, j, ny, k, g, h
Vegyes	p, b, m, f, v, (szájnyílás domináns, szélesség határozatlan) sz, z, c, dz (szájnyílás rugalmas, szélesség határozatlan)

3. táblázat Dominancia jellemzők a nyelv vízszintes helyzetére nézve

Domináns	t, d, n, r, l, ty, gy, j, ny, s, zs, cs, dzs, sz, z, c, dz
Rugalmas	magánhangzók
Határozatlan	p, b, m, f, v, k, g, h

A dominancia beállításai a paraméterek interpolációs szintjét határozzák meg. A további módosítások – pl. hosszú magánhangzóknál állandósult szakasz beiktatása – finomítják az artikulációt.

3 A természetesség javítása

A beszélő természetes fejmozgását, mimikáját hírolvasó bemondók felvételein tanulmányoztuk. Ennek nyomán álvéletlen mozgásokat, például visszafogott bólogatást, a fej enyhe oldalra billentését és átlag körül szóródó pislogási periódust alkalmaztunk. A prozódia tükröződése a fejmozgásban, illetve az arcmimikában nehezen algoritmizálható, így pl. a mondathangsúly kifejezése nehézségekbe ütközik. Az intonáció azonban felhasználható a szemöldök mozgatásának vezérlésére. A mondathangsúlynál is emelhető a szemöldök. A szemmozgást a fejmozgás korrigálására használjuk, hogy a tekintet ugyanarra a pontra szegeződjön, egyéb szemmozgatás kézi beavatkozást igényel. Dialógus rendszerekben a szerepváltást segíthetik a gesztusok, az értő figyelést a szemöldök emelésével jelezhetjük, bólogatással is viszszaigazolhatjuk figyelmes hallgatásunkat. Ezek a műveletek egyelőre manuálisan állíthatók be.

3.1 Az érzelmek kifejezése

A beszéd multimodális jellegéhez hozzátartoznak a gesztusok is. A testbeszéddel árnyaljuk mondandónkat, megerősítjük vagy éppen cáfoljuk verbális üzenetünket. Arcanimációs rendszerünkben az arckifejezések érzelmi töltését próbáltuk meg algoritmizálni és programozni. Az Ekman [8] által meghatározott hét érzelem közül választhatunk: semleges, haragos, ellenszenves, szorongó, boldog, szomorú, meglepett. Erre láthatunk példát a 4. ábrán.

4. ábra Ellenszenves és boldog arckifejezés

4 Összefoglalás és kitekintés

A cikk többéves kutató-fejlesztő munka eredményét ismerteti. A cél vizuális szövegfelolvasó rendszer kialakítása. A fejlesztés jelen fázisában az artikuláció dinamikus jellemzőinek további finomítását végezzük. A természetes vagy gépi beszédhez a szinkronizálás még nem teljesen automatikus, a következő feladatunk ennek megoldása. Több fejmodell működtetéséhez egységes leíró nyelvet kívánunk kialakítani, hogy egy új modellhez csak a modellfüggő átalakításokra legyen szükség. Igy az esetleges fejlesztéseket is csak itt kellene megvalósítani, nem minden modell vezérléséhez külön-külön. A fejlesztőrendszerünk a beszélő fej videó anyagát hosszadalmas számításokkal állítja elő, ami több órás feldolgozási időt is jelenthet. Jelenleg csak olyan alkalmazásokra gondolhatunk, ahol előzetesen rögzített üzeneteket jelenítünk meg. A nagy számításigény miatt szövegfelolvasásra is alkalmas rendszerünk jelenleg csak kötött szótáras alkalmazásokban használható. Reményeink szerint a real-time animáció a közeli jövőben szuperszámítógépek nélkül is megvalósítható lesz és ezzel a tényleges virtuális bemondói, felolvasói alkalmazások is megvalósíthatók lesznek.

5 Köszönetnyilvánítás

A kutatást az Informatikai és Hírközlési Minisztérium az ITEM program keretében támogatta 345 regisztrációs szám alatt.

Irodalomjegyzék

1. Massaro, D.W.: Perceiving Talking Faces. The MIT Press Cambridge, Massachusetts London, England (1998) 359-390
2. Bernstein, L.E., Auer, E.T.: Word Recognition in Speechreading. Speechreading by Humans and Machines. Springer-Verlag, Berlin Heidelberg, Germany, 1996, 17-26
3. Molnár József: A magyar beszédhangok atlasza Tankönyvkiadó, Budapest, 1986
4. Bolla Kálmán: Magyar fonetikai atlasz. A szegmentális hangszerkezet elemei Nemzeti. Tankönyvkiadó., Budapest. 1995.
5. Bolla Kálmán: Magyar hangalbum : A magyar beszédhangok artikulációs és akusztikai sajátságai MTA Nyelvtudományi . Intézet., Budapest. 1980.
6. Mátyás János.: Vizuális beszédszintézis. Diplomaterv Miskolci Egyetem 2003.
7. Czap, L.: Lip Representation by Image Ellipse. ICSLP 2000 Bejging, China, Proceedings Vol. IV. 93-96
8. Ekman, P., Friesen, W.: Facial Action Coding System Consulting Psychologists Press. Inc., 1978.

Talking Head

L. Czap[1], J. Mátyás[2]

[1] University of Miskolc, Institute of Electrical Engineering, Department of Automatisation
3515 Miskolc, Egyetemváros
czap@mazsola.iit.uni-miskolc.hu
[2] North Hungarian Regional Centre for Manpower Development and Retraining
3518 Miskolc, Erenyő u. 1.
matyasj@mail.erak.hu

Abstract

One of the aims of this assignment is to analyse the articulation of Hungarian speech sounds and, on the basis of this, to make a comprehensive study of how speech sounds can be visualized. In more specific terms, the assignment aims at determining, in a quantitative way, the shapes and movements of the lips to visualise the articulation of consonants and vowels. In this stage of the assignment an outstanding role has been given - besides modelling lip movements - to the modelling of characteristic tongue positions and tongue paths as well as to create the movement possibilities of brows, eyelids, eyes and the head. As a final outcome, a control algorithm has been developed, by means of which a three-dimensional head model can be operated to generate articulation of any required speech sounds corresponding to Hungarian text.

By analysing the articulation of Hungarian speech sounds, we have determined the lip shapes of Hungarian sounds. We have studied the different tongue positions and tongue paths, and the visibility of teeth as well. Similarly, we have studied the movement and deformation of the head, eyes, eyelids and brows on digitalized video clips, and for carrying out our assignment; we have used the relevant conclusions. We have compiled a collection of visemes which stores, as parameters, the visual snapshots together with the characteristic tongue positions. Target values have been assigned to the parameters, for each Hungarian viseme, based upon measurements made on human speakers. Parameter trajectories are modeled by means of dominance functions associated with each parameter and each viseme. A dominance function is characterized by three possible degree – dominant, flexible or undefined - so that coarticulation finally depends on the phonetic context.

For the operation of the three-dimensional head model, we have developed a processing programme which - considering the timings - is able to build up the basics of the controlling set necessary for animation. Controlling of the three-dimensional head model needs further developments. In this respect, the most important challenges will be to provide synchronization for the existing acoustic synthesizer and to create Real time animation possibilities providing the dialogue option as well.

A készülő Akadémiai nagyszótár számítógépes vonatkozásai

Pajzs Júlia

MTA Nyelvtudományi Intézet
Lexikográfiai és Lexikológiai Osztály
1068 Budapest Benczúr u. 33.
pajzs@nytud.hu

Abstract. The project for the Academic Dictionary of Hungarian is presented from computational point of view. The major steps are the following: collection of the 25 million running word Historical Corpus of Hungarian, lemmatization, disambiguation, user friendly retrieval interface (www.nytud.hu/hhc), frequency database of the entries, on-line compilation of the dictionary entries with the XML module of the Corel Office 2000 WordPerfect 9 program. Presentation of the TEI based DTD of the dictionary.

Bevezetés

Az Akadémiai nagyszótár munkálatai 1985-ben indultak újra. Mintául a francia *Trésor de la langue française* projektum szolgált, amelyben számítógépes korpuszra és hagyományos módon gyűjtött cédulákra támaszkodva készítették el a francia nyelv 1789 utáni történeti szótárát.

1. A korpusz összeállítása és rögzítése

A magyar szótár forrásanyagául szolgáló történeti korpuszba rögzítendő anyagot irodalomtörténészek és különféle területek szakértői válogatták ki számunkra. Az anyag kijelölésével párhuzamosan megindult a próbaszövegek bevitele (kezdetben Commodore 64 típusú számítógépeken). Először saját kódrendszert alakítottunk ki a rögzítésre, az ékezetes és történeti karaktereket az angol ábécé betűiből és számok kombinációjából álló kódrendszerrel jelöltük (ún. Prószéky kód: á=a1, ö=o2 stb.). Amint megjelent a TEI SGML szabvány, áttértünk a korpusz TEI alapú kódolására, az ékezetes és történeti karaktereket azonban – a biztonságos hordozhatóság és egyértelmű konvertálhatóság érdekében – továbbra is Prószéky kódban tároljuk.

2. Morfológiai elemzés

A munkálat elindításakor elhatároztuk, hogy egy morfológiai elemző program segítségével könnyebben kezelhetővé, lekérdezhetővé tesszük az anyagot. Az elemző program terveit Prószéky Gábor készítette. A megvalósításhoz felhasználtuk Elekfi

László: *Szókincsünk nyelvtani alakrendszere* című művét. Az elemző program első változatát magam készítettem [18]. Később Tihanyi László csatlakozott a munkacsoporthoz, és ő folytatta a program írását, amelynek továbbfejlesztett változata HUMOR morfológiai elemző program néven vált ismertté, és a HELYES-E? alapjává.

Az elemző használata lehetővé tette, hogy ne pusztán szövegszavakat, szóalakokat keressünk a korpuszban, hanem lexémákat is. Így a lexikográfusoknak nem kell a szavak minden lehetséges toldalékolt alakját egyesével kikeresgélni, egyszerre lekérdezhetik az *alszik*, a *bokor* vagy a *hó* szó összes toldalékolt alakját.

Míg a mai szövegekre a HUMOR megfelelő hatékonysággal alkalmazható, a régiekre természetszerűleg lényegesen kevésbé. Ezért egy olyan heurisztikus algoritmusokkal dolgozó programot fejlesztettünk [12], [13], [14] amely az esetek jelentős részében lehetővé teszi a régies alakok helyes elemzését is.
Az eljárás lényege a következő:

A HUMOR program elemzi a szövegszavakat.

- Ha egy szót nem sikerült elemezni, a heurisztikus program átalakítja, majd újra elemezni próbálja azt.
- Ha az átalakított változat elemzése sikeres, a felismert, vagy felismerni vélt alakot őrizzük meg.

Az átalakításokat a történeti szövegekben megfigyelhető szabályszerűségek alapján végezzük el. A program első változata csaknem 30 százalékkal növelte a lemmatizálható szövegszavak mennyiségét.

3. Egyértelműsítés

Az 1990-es évek közepén fejlesztettem és teszteltem egy szabályalapú egyértelműsítő programot [18],[19]. Bár a történeti szövegek sajátosságai miatt az eredmények távolról sem voltak olyan jók, mint amilyenekről akár a nemzetközi, akár a hazai korpusznyelvészeti és nyelvtechnológiai kutatások számot adnak, a korpusz használhatóságát, lekérdezhetőségét jelentősen javította az eljárás alkalmazása.

4. Címszólista

Az elemzett és egyértelműsített korpusz alapján készítettem egy olyan címszógyakorisági listát, amely a gyakorisági adatokon túl a szó első és utolsó előfordulásának idejét is tartalmazta. Ebből nem csak a korpusz címszóállományának gazdaságáról kaphattunk képet, hanem a szavak időbeli eloszlásáról is (Kiss G. – Pajzs 2000, Pajzs 1997). A Morphologic Kft.-ben Tihanyi László készítette el a cimszójegyzék Mobidic felületű változatát.

5. Lekérdezés

A korpusz lekérdező programját, amely külső felhasználók részére is szabadon hozzáférhetővé teszi a korpuszt (www.nytud.hu/hhc) Váradi Tamás készítette [17]. A

lekérdezőnek korábban volt egy `telnet` alatt működő változata is, ez azonban ma már csak a belső felhasználók számára használható, biztonságtechnikai okokból. Lexikográfiai felhasználásra a lekérdezőnek kissé módoított változatát fejlesztettük ki, a `telnet` alattt működő változatot – Váradi Tamás programjának felhasználásával – fejlesztettem, ez az elemzett és egyértelműsített korpuszon működik, és a PAT (Open Text) lekérdező programot használja motorként. Egy másik, szintén a lexikográfusok speciális igényeit kiszolgáló programot Nagy Viktor készített számunkra, ennek lekérdező modulja a stuttgarti egyetemen fejlesztett Corpus Workbench program. Egyelőre ez is csak a belső felhasználók számára érhető el.

6. A szótári adatbázis készítése

A szócikkeket közvetlenül XML formában készítjük, a WordPerfect 9. XML szerkesztő moduljának felhasználásával. A szótár DTD-jét, és az ehhez tartozó XML applikációt – a TEI standard javaslatainak figyelembevételével – magam készítettem, az utóbbi időben Mártonfi Attila munkatársam tartja karban.

6.1 A korpusz Dokumentum típus definiciója

```
<!DOCTYPE dic
[
<!ELEMENT dic       (fileName, (entry | entryxr)+)>
<!ELEMENT entry  ((remark?, head, (sense | sengr)*,
xref*), compby?)>
<!ELEMENT head   (lemma, usg*, gramgrp?, (usgvar?,
variant)*, usg*, xref*, freq?)>
<!ELEMENT lemma  (#PCDATA | hom)*>
<!ELEMENT hom    (#PCDATA)>
<!ELEMENT variant (#PCDATA)>
<!ELEMENT gramgrp (subc*, pos*, gov?, lbl*)>
<!ELEMENT pos    (#PCDATA)>
<!ELEMENT subc   (#PCDATA)>
<!ELEMENT lbl    (#PCDATA | mention)*>
<!ELEMENT mention (#PCDATA |hint | hom)*>
<!ELEMENT usg    (#PCDATA)>
<!ELEMENT sense  (sennu?, (mainsens | sumsens)?,
subsen*)>
<!ELEMENT sennu  (#PCDATA)>
<!ELEMENT mainsens (usg*, gramgrp?, reflex?, usg*,
(def|defrep)*, dom*,hideg | eg)*, (re* | coll*))>
<!ELEMENT sumsens (usg*, defsum, coll*)>
<!ELEMENT sengr  (sgrnu, gramgrp, xref*, sense* )>
<!ELEMENT sgrnu  (#PCDATA)>
<!ELEMENT def(#PCDATA | gloss | hint | abbr | tr |
        mention| syn| dom )*>
<!ELEMENT defrep  (#PCDATA | gloss | hint| abbr| tr |
        mention | syn| dom )*>
```

```
<!ELEMENT defsum  (#PCDATA | gloss | hint | abbr | tr |
mention | syn | dom)*>
<!ELEMENT hint    (#PCDATA | usgphr?)*>
<!ELEMENT hinteg  (#PCDATA | hide )*>
<!ELEMENT coll    (usgphr*, (ph | hint)*, ((subd*) |
usgphr*, (def|defrep|defsum), (eg | hideg)*)))>
<!ELEMENT subd      (sdnu,usg*, (def|defrep), (eg |
hideg)*)>
<!ELEMENT eg        (cit,bibl)>
<!ELEMENT hideg     (cit,bibl)>
<!ELEMENT hide      (#PCDATA | hinteg | ref | ref2 | ph |
abbr)*>
<!ELEMENT bibl     (wdate, (author | pubTitle), id, p)>
<!ELEMENT xref    (xrtype?,xr*)>
<!ELEMENT xr      (#PCDATA | hom)*>
<!ELEMENT xrtype  (#PCDATA)>
<!ELEMENT wdate   (#PCDATA)>
<!ELEMENT cit     (#PCDATA | hinteg | ref | ref2 | ph |
abbr | hide )*>
<!ELEMENT ref     (#PCDATA | abbr )*>
<!ELEMENT ref2    (#PCDATA)>
]>
```

6.2 Az egyes tagek jelentése

<abbr>	a példamondaton belül rövidített vagy tildével helyettesített szó kiegészített, feloldott része
<author>	szerző, magyar fordító
<bibl>	bibliográfiai adatok egysége
<cDate>	a szócikk írásának időpontja
<cit>	idézet
<cName>	a szócikkíró neve
<coll>	értelmezett szókapcsolatok egysége
<compby>	a szócikkre vonatkozó információk blokkja
<deduced>	szóadatok nélküli, szókapcsolatból kiemelt címszó, paradigmatikus alakból elvont alakváltozat szögletes zárójelben
<def>	értelmezés
<defrep>	helyettesítő értelmezés csúcsos zárójelben
<defsum>	összefoglaló értelmezés
<dictions>	szótári hivatkozás (példamondat helyett, ill. szócikk végi blokkon belül)
<dom>	fogalomköri besorolás (az értelmezés része is lehet)
<eg>	a példamondat egysége
<entry>	önálló szócikk, ill. szócikkfejes utaló szócikk
<gloss>	az értelmezés csúcsos zárójeles kiegészítő része
<gov>	vonzat

<gramgrp>	grammatikai információk blokkja
<head>	szócikkfej
<hide>	rejtés (a teljes idézet vagy annak egy része rejtve)
<hideg>	a teljes rejtett példák (idézet + bibliográfia hivatkozás is) egysége
<hint>	az értelmezés kerek zárójeles vagylagos része, ill. az értelmezett szókapcsolat kimaradható része
<hinteg>	a példamondaton belüli szöveges kiegészítések és kihagyások szögletes zárójelben
<hom>	homonima indexszáma
<id>	a forrás kódszáma
<lbl>	nyelvtani kiegészítés a szófaj után kerek zárójelben
<lemma>	címszó
<mainsens>	adatolt főjelentés
<mention>	az <lbl>-en, ill. a <def>-en belüli kurzív rész (pl. `-t raggal hsz-szerűen; ill. 'abcűg kiáltással lehurrog vkit'`),
<nCorp>	korpuszbeli adatok száma
<nElse>	CD-adatok száma
<nSlip>	cédulák száma
<orauthor>	eredeti szerző (fordításoknál, a szócikkbe nem kerül be)
<p>	oldalszám
<ph>	az értelmezett szókapcsolat főváltozata adatként + a példamondat kiemelt részeként
<pos>	szófaj
<pubDate>	megjelenés éve (a szócikkbe nem kerül be)
<pubTitle>	a mű címe
<re>	bokrosított szócikk alcímszava
<ref>	a címszó előfordulása a példamondatban, kivéve értelmezett szókapcsolat részeként
<reflex>	igéből és *magát* tárgyból álló szerkezet (önálló jelentésben)
<remark>	a szócikkírás közbeni megjegyzések, figyelmeztetések, itt emeljük ki a filológiai ellenőrzendőket
<sdnu>	aljelentés **a)**, **b)** stb. betűjele
<sengr>	szófaji blokk
<sennu>	jelentésszám
<sense>	a jelentés blokkja
<sgrnu>	a szófajt jelölő római szám
<sources>	szócikk végi szótári hivatkozások blokkja
<status>	a szócikk állapota
<subc>	a szófajt megelőző szófaji kiegészítők (ts, tn)
<subd>	aljelentés egysége (ha ugyanaz a frazéma több jelentésben is használatos)
<subnu>	jelentésárnyalat száma
<subsen>	jelentésárnyalat
<sumsens>	adatok nélküli összefoglaló jelentés
<syn>	szinonima az értelmezésben
<tr>	az értelmezés kerek zárójeles kiegészítő részei (pl. latin növénynév)

<tra>	a ford. rövidítés egysége
<type>	a szótári hivatkozások **Vö.** rövidítése
<usg>	lexikai minősítés, ha a teljes szócikkre, ill. egy szófajra, jelentésre vagy frazéma aljelentésére vonatkozik
<usgphr>	az értelmezett szókapcsolatra, ill. a szólásra vonatkozó lexikai minősítés
<usgvar>	egy alakváltozatra vonatkozó lexikai minősítés
<variant>	alakváltozat a szócikkfejben
<vol>	kötetszám (a szócikkben nem jelenik meg)
<wdate>	keletkezés éve
<xr>	az a címszó, amelyre utalunk, ill. a szótári hivatkozásokban kiírandó címszó
<xref>*	összetételi és frazeológiai utalások blokkja (szócikk végén, belsejében, ill. csak ilyen szócikkfejes utalások szófaji minősítése után)
<xrtype>	az utalás típusának rövidítése

ágyszék fn

1. *(rég)* 'kanapé, pamlag': mind-a'-hárman az *Ágyszékre* le-ültenek (1790 Dugonics András C1468, 111) | {Ereszkedj-le édesem erre az *ágyszékre* (1793 Magyar játék-szín C2989, 22)} | Egy/zerre felugrott [Dorottya] az *ágy/zékröl* (1803 Csokonai Vitéz Mihály 1800069016, 48).

2. *(nyj)* 'lábazaton álló egyszerű deszkaágy': Az ágynak legősibb formája: az *ágyszék*, mely négyszegletű lábakból, deszkákból összerótt, kezdetleges tákolmány (1922 A magyar nép művészete CD07).

Vö. CzF.; ÚMTsz.

Szócikkíró: *Kéthely Anna*; első változat kelte: 2003. 01.

Szerkesztö: *Ittzés Nóra*; szerkesztés kelte: 2003. 03.

Cédulák száma: 30. Korpuszbeli adatok száma: 1.

A szócikk állapota: 3.

```
<entry><head><lemma>ágyszék</lemma>
<gramgrp><pos>fn</pos></gramgrp></head>
<sense><sennu/><mainsens>
<usg>rég</usg>
<def>kanapé, pamlag</def>
<eg><cit>mind-a'-hárman az <ref>ágy-székre</ref> le-u24ltenek </cit>
<bibl><wdate>1790</wdate><author>Dugonics András</author>
<id>C1468</id><p>111</p></bibl></eg>
<hideg><cit>Ereszkedj-le édesem erre az <ref>ágyszékre</ref></cit>
<bibl><wdate>1793</wdate><pubTitle> Magyar játék-szín </pubTitle>
<id>C2989</id><p>22</p></bibl></hideg>
```

Fig 1. Egy mintaszócikk kinyomtatott formában és XML változatának eleje

A <hideg> cimkével jelölt példamondatok csak a szótár elektronikus változatában fognak megjelenni. Más esetekben csak a példamondat egy részét rejtjük el a nyomtatáskor, ezeket a <hide> taggal jelöljük meg. Összességében a példamondatok bő egyharmada csak az adatbázis változatban lesz látható.

A készülő szótári adatbázis használata

Az XML változatban készülő szócikkekből nem csak a nyomtatott és az elektronikus változat állítható elő könnyedén, már munka közben is segíti, hogy több szempontból lekérdezhessük, ellenőrizhessük szócikkeinket. A szócikkek strukturális és mennyiségi ellenőrzését és lekérdezését pillanatnyilag Perl programok segítségével oldjuk meg, amelyek ACCESS adatbázisba exportálják a legfontosabb adatokat, ezekből azután a legkülönbözőbb lekérdezéseket generálhatjuk (pl. a példamondatok időbeli eloszlása, leggyakrabban idézett szerzők, melyik mondatokban szerepel Kádár János, vagy 1956 stb.) Ezen adatbázisok segítenek a munkatársak teljesítményének pontos, naprakész mérésében is.

Referenciák

[1] B. Lőrinczy É.-Gerstner K.: Lehet-e végre a magyar nyelvnek nagyszótára *Magyar Tudomány* 105 (1998): 261–71.

[2] Csengery K.-Ittzés N.(eds): Mutatványok az Akadémiai nagyszótárból MTA Nyelvtudományi Intézet, Budapest, 2002.

[3] Elekfi L.: Nagyszótári tervek és lehetőségek I–II *Magyar Nyelv* 93 (1997): 183–99, 296–311.

[4] Elekfi L.: Melléklet a Nagyszótári tervek és lehetőségek c. közleményhez: *út Magyar Nyelv* 94 (1998): 235–53.

[5] Elekfi L.: Nagyszótári tervek és lehetőségek III. *Magyar Nyelv* 94 (1998): 374–8.

[6] Elekfi L.: Mit tartalmaz a Szókincsünk nyelvtani alakrendszere c gyűjtemény? *Magyar Nyelv* 93 (1997): 63–8.

[7] Elekfi L.: A Magyar ragozási szótár és „Szókincsünk nyelvtani alakrendszere (1997): 213–21.

[8] Elekfi L.: Eltérő toldalékokban mutatkozó jelentéskülönbségek *Magyar Nyelvőr* 122 (1998): 305–17.

[9] Elekfi L.: Semantic differences of sufficial alternates in Hungarian. *Acta Linguistica Hungarica* 47 (2000): 145–77.

[10] Elekfi L.: Homonimák felismerése toldalékos alakok alapján. *Magyar Nyelvőr* 124 [2000]: 146–63.

[11] Gerstner K.: Cédulák és fájlok – A Magyar akadémiai nagyszótár alapjairól. In: Kiefer F.–Gósy M. (eds): *Helyzetkép a magyar nyelvtudományról* MTA Nyelvtudományi Inézet, Budapest, 2000, 35–43.

[12] Kiss Gabriella .– Pajzs J.: An attempt to develop a lemmatiser for the Historical Corpus of Hungarian. *Proceedings of CL 2001*. University of Lancaster, (2001),

[13] Kiss, Gabriella – Kiss, Margit – Pajzs, Júlia: Normalisation of Hungarian Archaic Texts Proceedings of COMPLEX 2001, University of Birmingham, (2001), pp. 83-94.

[14] Kiss L.-Pajzs J.: A magyar irodalmi és köznyelv nagyszótára (1533–1990) *Magyar Nyelv* 85 (1989): 129–36.

[15] Pajzs J.: Creating a Historical Dictionary of Hungarian with the Aid of Computer. In: *T. Magay–J. Zigány: BUDALEX '88 Proceedings.* Akadémiai Kiadó, Budapest, (1990), 559–63.

[16] Pajzs J.: Réalisation assistée par ordinateur de grands dictionnaires français et hongrois. *Cahiers d'études hongroises 3/91 Centre Interuniversitaire d'Études Hongroises Université Paris III.* Institut Hongrois de Paris, 47–54.

[17] Pajzs J.-Váradi T.: A magyar irodalmi és köznyelv nagyszótárának korpusza a HUNGARNET közösség számára. *A Networkshop '97 konferencia anyaga.* CD Budapest, NIIF, (1997).

[18] Pais J.–Pajzs J.: Using local rules for disambiguation of Homographs in Hungarian corpora. *Proceedings of the EURALEX '98 Conference.* University of Liège, 1998, 239–48.
[19] Pajzs J.: Synthesis of results about analysis of corpora in Hungarian. *Linguistiae Investigationes* XXI/2. John Benjamins, Amsterdam, (1997), 349–65.
[20] Pajzs, J.: Making Historical Dictionaries with the Computer. *Proceedings of EURALEX 2000,* University of Stuttgart, (2000), 249–59.

A szógyakoriság és helyesírás-ellenőrzés

Halácsy Péter[1], Kornai András[2], Németh László[1], Rung András[3], Szakadát István[1] és Trón Viktor[4]

[1] Budapesti Műszaki és Gazdaságtudományi Egyetem, Média Oktatási és Kutató Központ
{halacsy,szakadat,rung}@mokk.bme.hu
[2] MetaCarta Inc.
andras@kornai.com
[3] Budapesti Műszaki és Gazdaságtudományi Egyetem, Kognitív Tudományi Központ
rung@itm.bme.hu
[4] International Graduate College of Language Technology and Cognitive Systems
Saarland University – University of Edinburgh
v.tron@ed.ac.uk

Kulcsszavak magyar webkorpusz, gyakorisági szótár, helyesírás-ellenőrzés, lexikográfia

Absztrakt A Szószablya projekt márciusában indult az Informatikai és Hírközlési Minisztérium támogatásával, a Budapesti Műszaki és Gazdaságtudományi Egyetemen működő Média Oktatási és Kutató Központ vezetésével. A projekt egyik kiemelt célja, hogy a magyar weboldalak szövegtartalma alapján egyedülálló teljességű magyar szógyakorisági szótárat, illetve ehhez kapcsolódó nyitott forráskódú alkalmazásokat készítsen. A cikkben röviden bemutatjuk a gyakorisági szótár készítésének menetét (§1), valamint a beépített szótárat használó alkalmazások, mint a helyesírás-ellenőrző szóanyagának hatékony bővítésére kidolgozott gyakorisági információn alapuló technológiát. A cikkben megmutatjuk, hogy a helyesírás-ellenőrzők pontosságát elsősorban a tőtáruk mérete határozza meg (§2), azonban a tőtár bővítése a Zipf törvény miatt egyre kisebb pontosságnövekedéssel jár (§3).

1. A magyar web és feldolgozása és a Szószablya gyakorisági szótár

A letölthető magyar web durva becslés alapján 20–25 millió oldalból áll, amelynek jelentős része szövegtartalom nélküli, idegen nyelvű, vagy duplikált oldal; egyes oldalak folyamatosan változnak, így a letöltését egy szűkebb időintervallumra kell korlátoznunk. A SzóSzablya első korpusza 2,4 millió weboldalt tartalmazó lar0 2002 decemberében készült el, míg a második, jóval teljesebb, 18 millió oldalt tartalmazó gul0 2003 őszét tükrözi. A letöltött nyers korpuszok szűrését és feldolgozását itt csak röviden ismertetjük (részletesen ld Németh 2003).

A lapformátumok, a szövegtartalmak és a karakterkódolás normalizálását a Szószablya keretében kifejlesztett Hunnorm alkalmazás segítségével végeztük el. Ezután a szövegtartalom alapján történő duplikátumszűrésre került sor, amelyet a magyar ékezetes karaktereket nem tartalmazó szövegek kizárása követett. A szógyakorisági szótár elkészítéséhez a korpusz szövegeit tokenizáltuk a Huntoken program segítségével.[5]

A gyakorisági számítások jelentős torzulását okozhatja, hogy az azonos szolgáltatóhoz tartozó weboldalak egységes automatikus generált címszavai a tényleges gyakoriságuknál jóval nagyobb számban jelennek meg a gyűjtésben. Emiatt az oldalakat csak az első mondatzáró pont utántól vettük figyelembe, és az így kapott halmazon újabb tartalomalapú duplikátumszűrást végeztünk. A szűrések után a korpuszok az oldalak számát tekintve rendre 46, illetve 75%-kal csökkentek.

Az így kapott szöveganyag a nyers eredetihez képes jelentősen javult minőségileg, a magyar webköznyelvhez jobban közelítő gyakorisági adatok kinyeréséhez azonban egy további kritériumot alkalmaztunk: a Hunspell helyesírás-ellenőrző által elutasított szavak arányát. Ha ugyanis feltételezzük, hogy egy szöveg helyesírása egységes, akkor a helyesírás-ellenőrző által helyesebbnek ítélt oldalakból nyert szóalakok is nagyobb valószínűséggel lesznek helyes szavak, vagyis a szóanyag minőségi előszűrése elvégezhető a helyesírásellenőrző lefedettségétől függetlenül. A Szeged Korpusz állományain elvégzett előzetes helyességvizsgálat alapján a tisztított gyakorisági szótárunk számára a 4% százalékos felső Hunspell-hibaküszöböt tartottuk megfelelőnek. Ez a kritérium a szűrt weboldalak további 60%-át szűrte ki.

Az így nyert anyag, a `webO` 433238 weboldalból áll, $N = 113385165$ szövegszót (token) és $V(N) = 4527456$ szóalakot (típus) tartalmaz. Becslésünk szerint a `webO` típusainak 80%-a helyes magyar szóalak, ami lényeges javulás a teljes webkorpuszhoz képest, amelynek szótípusain ez az arány 50% alatt van. Még szembetűnőbb a hibás szövegszavak előfordulásának csökkenése: ez becslésünk szerint 15%-ról 2,5%-ra csökkent. A teljesen automatikus módszerekkel nyert `webO` alapján elkészített első Szószablya szógyakorisági szótár elérhető a projekt honlapján.

A gyakorisági szótárak jelentősége felbecsülhetetlen mind a nyelvtechnológiában mind nyelvészeti kutatásokban, felhasználási területei számosak a fordítástámogatástól a pszicholingvisztikai kísérletekig. Az alábbiakban egy speciális területre koncentrálunk: azt elemezzük, hogy a gyakorisági információ milyen módon segítheti a a nyelvtechnológiai alkalmazások beépített szótárának hatékony bővítését, vagyis az alkalmazás lefedettségének a növekedését. A technológiát a helyesírás-ellenőrző alkalmazáson szemléltetjük.

[5] Elsősorban tulajdonnevek és a rövidítések későbbi hatékonyabb feldolgozása érdekében a szövegek mondatra bontását is elvégeztük.

2. Helyesírásellenőrzés és pontosság

A helyesírás-ellenőrzés alapvető célja, hogy segítségével a szövegben található hibák számát csökkentsük. Egy helyesírás-ellenőrző program tényleges hibáját az határozza meg, hogy (i) a szövegben lévő helyes szavak közül hányat utasít el, és (ii) a helytelen szavak közül hányat fogad el. Ha tehát egy szöveg ellenőrzésekor minden 100 szó közül az ellenőrző csak egyszer hibázik (akár (i), akár (ii) típusú hibát vét), akkor azt mondjuk, hogy az ellenőrző hibája h=1%, pontossága $1-h$= 99%. Méréseinket tokenekre, nem pedig típusokra alapozzuk. Ennek a módszertani döntésnek a hátterében az a megfigyelés áll, hogy a gyakori alakokon vett pontosság sokkal fontosabb a ritka, egzotikus szóalakok hibátlan kezelésénél. A hibaszázalék (h) meghatározásakor tehát akkor nyerünk a felhasználó gyakorlati tapasztalaival egybevágó eredményt, ha az egyes szóalakokat gyakorisági súlyuknak megfelelően vesszük számításba[6]

A helyesírásellenőrző működését a továbbiakban automatikus javítási módban képzelhetjük el, vagyis amikor az ismeretlen szavakra szerkesztési távolság és gyakoriság alapján tesz helyettesítést. Azokra az alakokra, amelyekre nincs megbízható cserejavaslat, elfogadásra kerülnek. A pontosságot így három tényező befolyásolja döntően: a helyesírás-ellenőrző lefedettsége l, belső hibája b, és a maradványhiba m. A lefedettség egyszerűen azon tokenek aránya a szövegben, amelyekre nézve a helyesírás-ellenőrző képes javaslatot tud tenni (maga a szóalak vagy egy ahhoz közeli karakterfüzér szerepel a szótárában). A belső hiba magából a javaslattevésből adódó hiba, a maradványhiba pedig a javaslattevésből kimaradt és így elfogadott szavak között szereplő helytelen alakok aránya. Az összevont h hiba tehát a belső hiba és a maradványhiba lefedettséggel súlyozott átlaga:

$$h = lb + (1 - l)m \tag{1}$$

A lefedettséghez csak a javaslattevések és az elfogadott szavak számát kell tudnunk, l mérése tehát triviális. m mérése azonban jóval összetettebb feladat, amely nem automatizálható. Az egyszerűség kedvéért feltesszük, hogy m felülről becsülhető az egész szövegben levő hibától, vagyis m l-lel nem növekszik.[7] Akkor érdemes egyáltalán helyesírásellenőrzőt használni, amikor az aktív ellenőrzés belső pontatlansága kisebb, mint a szöveg hibája ($b < m$), ekkor azonban h optimalizálása l növelésével, vagyis a tőtár növelésével érhető el. Felvethető a kérdés: mi a leghatékonyabb módszere korpuszok alapján történő szótárbővítésnek,[8] vagyis milyen módszerrel garantálható, hogy a legkisebb befektetéssel a hibaszázalék legnagyobb növekedését érjük el?

[6] A tokenszintű hibaszámítás a kontextuális információkat is felhasználó helyesírásellenőrző esetén még fontosabb, így ez az eljárás lehetővé teszi az eredmények összevetését.

[7] Tudjuk, hogy a szövegre jellemző hibaérték nagyban függ a szöveg típusától: *ekezetteien irasmodban* akár a 30-40%-ot is elérheti, átlagos szövegben 5-6% körüli, gondozott szövegben tipikusan 0.1% alatt marad.

[8] A helyesírás-ellenőrző szótárbővítése természetesen részben kivetelezhető új tövekben gazdag anyagok (szótárak, helységnévtárak, cégjegyzékek, stb.) átvételével is, de a kézi átnézésre még szótáraknál is szükség van a sajtóhibák miatt.

3. Szógyakoriság és a lefedettség növelése

A naiv helyesírás-ellenőrző rendszerek alapját a helyes alakok gyakoriság szerint rendezett rögzített listája adja. A magyarban $l > 0.5$ eléréséhez már az első néhány ezer alak figyelembevétele is elégséges: ha ezeket kézzel átnézzük akkor $b \approx 0$ garantálható,[9] így (1) alapján $h < m/2$ minden nehézség nélkül elérhető.

Az alábbiakban a kvantitatív állításokat a Magyar Webkorpusz két változatán is illusztráljuk, az adatokat `web0` gyűjtés mellett összehasonlításképpen a $N = 670076633$ szövegszót és $V(N) = 15057395$ típust tartalmazó szűretlen `lar1` gyűjtésre is megadjuk.

Az 50%-os lefedettség garantálásához a 2913 (`lar1`), illetve 6486 (`web0`) szóalak listába vétele szükséges. Még $l > 0.666$ (tehát $h < m/3$) is csupán 15 ezer (`lar1`), illetve 24 ezer alak (`web0`) kézi átnézését igényli. A naiv módszer korlátját az adja, hogy az alacsony tokengyakoriságú szavakból típus szerint nagyon sok van. Legyen a mintában pontosan a k-szor előforduló típusok száma $V(k, N)$, tehát $N = \sum_{k=1}^{\infty} kV(k, N)$ és $V(N) = \sum_{k=1}^{\infty} V(k, N)$. Általános tapasztalat (ld. pl. Baayen 1996), hogy ha $N > 10^6$, akkor $V(N)$ domináns tagja $V(1, N)$, a típusok több mint fele olyan, hogy az egész szövegben csak egyszer fordul elő, vagyis ún. hapax legomenon. A `lar1` korpuszban $V(1, N) = 8133805$, a `web0`-ban $V(1, N) = 2567665$, vagyis a szóalakoknak rendre 54.0, illetve 56.7 százaléka hapax, bár gyakorisággal súlyozott arányuk csak rendre 1.21%, illetve 2.26%. A hapaxok kézi átnézése ezért lehetetlen, vagyis a lefedettség ezzel a módszerrel nem vihető 98% fölé, vagyis a h alsó korlátja $m/50$. Gyakorlatilag még a kétszer és háromszor előforduló szavak is komoly akadályt jelentenek: példáinkban ezek együttes gyakorisága mindössze 7.2% (`lar1`), illetve 4.2% (`web0`), a lista bővítéséhez azonban így is alakok millióit (8.1, illetve 4.7 millió) kellene kézzel átnézni! Gyakorlatban tehát a naiv módszer alsó hibahatára inkább $h \approx m/20$; ez egybevág a gyakorlati tapasztalattal, miszerint az átlagos (5-6% hibaarányú) nyers szövegből a naiv elven működő helyesírás-ellenőrzők lényegesen jobb (0.3% hibájú), azonban a gondozott szöveg minőségi követelményeit (0.1% alatti hiba) el nem érő javított változatot állítanak elő.

A tisztán izoláló nyelveknél, mint pl. a vietnami, a naiv módszer teljesen kielégítő, hiszen ezekben a nyelvekben a helyesírás ismerete nem jelent többet, mint az egyes szavak helyes ismerete[10] A komplexebb morfológiájú nyelveknél, mint amilyen a magyar, a helyes szóalakok ismerete nem merül ki a szótövek és a ragok ismeretében, hiszen ezek konkatenációja csak a morfológia szabályainak figyelembevételével eredményez helyes szóalakot. A helyesírás-elemzőbe tehát be kell építeni nemcsak a töveket és a ragokat, hanem a morfológiát is.[11] A hiba az ilyen rendszer esetén is a lefedettség lineáris függvénye, utóbbi viszont három komponensre bontható: a tő , a ragok, és a hasonulási szabályok lefedettsége.

[9] Szigorú értelemben b nem nulla, hiszen egy biztosan jó szótípus adott környezetben való előfordulása lehet, hogy hibásan kerül elfogadásra. Az ilyen hibákat elhanyagolhatónak tekintjük

[10] A megfontolás lényegében változtatás nélkül átvihető az olyan nyelvekre is, mint az angol, ahol az azonos tőhöz tartozó alakok száma kicsi.

[11] Ezt látjuk a legtöbb magyar helyesírás-ellenőrzőnél, pl. Hunspell, Helyes-e, Lektor.

A nem-rekurzív, tehát tételesen felsorolandó kombinációk száma nem túl nagy (pl. Veenker (1968) 3020 ragkombinációt vesz lajstromba, a Hunspell jelenleg 4936 főnévi, 4041 melléknévi, és 59 igei alakkal dolgozik).[12] Miután ezek kézzel könnyen ellenőrizhetők, feltehető, hogy a rendszer lefedettsége ebben a ragkombinációk tekintetében 100%. Érettebb rendszer esetén még a morfofonológia szabályainak teljes ismerete is elvárható, így a belső hiba legfontosabb forrása az lehet, hogy a tőtárban egyes elemek hibás morfológiai információval szerepelnek. A morfológiai elemzést használó helyesírás-ellenőrzőknél a tőtár bővítése nem egyszerűen az új tövek felvételét jelenti, hanem előfeltételezi a tövek morfológiai osztályozását is, így a fentiekkel ellentétben $b = 0$ nem garantálható. A hiba csökkentésének alapvető módszere itt is a mohó algoritmus: először bevesszük a leggyakoribb alakokat akár elemzetlenül is, különösen ha elemzésük túlságosan komplikálná a morfonológiai szabályrendszert (hiszen ennek hibáját 0-n akarjuk tartani), és csak akkor lépünk tovább a $T + 1$-edik tőhöz, ha az első T tő már minden gyakoribb alakot lefed. Ezzel az eljárással olyan alakokat is lefedünk, amelyek önmagukban nem kerülnének be (vagy mert hapaxok, vagy akár elő sem fordulnak az adott mintában), de mint gyakoribb tövek ritkább ragokkal való kombinációi most mégis elérhetővé válnak. Ilyen pl. a *decembereinknek* alak, amely kétségkívül jólformált magyar szó, még a 670m szövegszavas mintánkban sem fordul elő (és a naiv alak-gyakorisági megfontolások alapján soha nem is kerülne be a listába), így viszont a *december* gyakoribb tőszármazékai, valamint a produktív toldalékolási minták jóvoltából elfogadásra kerül.

Az általunk használt szótárbővítési eljárás tehát három lépésre bomlik: először gyakoriság szerint sorba rendezzük a szóalakokat, másodszor vesszük a még hiányzó leggyakoribb alak tövét. Ezt automatikusan állítjuk elő, a projekt keretében kifejlesztett Hunstem tövezővel, amely Hunspell-lel azonos morfológiai szabályrendszerrel dolgozik. Végül megállapítjuk a tő helyes besorolását. A mohó algoritmus egyre kisebbeket tud csak kivenni a maradékból, és a csökkenés mértéke is jól megbecsülhető. A szóalakok gyakoriságát első közelítésben Zipf törvénye adja meg: az r-edik alak valószínűsége $1/r^B$-vel arányos:[13]

$$p_r \sim 1/r^B \tag{2}$$

Miután az alakok valószínűsége a tő valószínűségének konstansszorosa (Kornai 1992) az összefüggés a tövekre is érvényes. Ennek alapján tehát a mintában pontosan k-szor előforduló tövek arányára a következő adódik (a levezetés részleteit ld. Kornai 1999):

$$V(k,N)/V(N) = 1/k^{1+1/B} \tag{3}$$

Éppen a helyesírási hibák miatt, ez még további korrekciót igényel, de $k = 1$-re áll, hogy $V(1,N)/V(N) > 0.5$. A magasabb tagok igen jól illeszkednek a Zipf törvény által jósolt (3) értékekhez: az alábbi táblázat $k \leq 10$-re mutatja $V(k,N)$ mért, illetve a $V(N)/2k^{1.8}$ formulával becsült értékeit, valamint a becslés relatív pontosságát a `lar1` és a `web0` korpuszokra:

[12] Bár a ragok és rag-kombinációk száma elvileg végtelen, a rekurzív esetek (túlzófok *legeslegesleges*..., birtokos *éééé*...) viszonylag egyszerű szabályokkal kezelhetők.

[13] A B Zipf konstans a magyarban 5/4 körül van (ld. Füredi et al 2003).

	lar1			web0		
k	$V(k,N)$	$V(N)/2k^{1.8}$	b/m	$V(k,N)$	$V(N)/2k^{1.8}$	b/m
1	8133805	7528697	0.925606	2567665	2263728	0.881629
2	2393113	2162050	0.903447	632426	650085	1.02792
3	1006086	1042081	1.03578	280327	313333	1.11774
4	628829	620886	0.987369	167517	186688	1.11444
5	385044	415503	1.0791	112747	124933	1.10809
6	297081	299259	1.00733	82215	89981.2	1.09446
7	211134	226748	1.07395	63211	68178.6	1.07859
8	175753	178303	1.01451	50017	53612	1.07188
9	137100	144239	1.05207	41053	43369.8	1.05644
10	115697	119322	1.03133	34623	35877.7	1.03624

Zipf törvénye szerint, ha lista bővítését a rangsor r-dik tagjánál abbahagyjuk, a lefedetlen rész aránya, $1 - l = \sum_{i=r}^{\infty} 1/r^B$. Ebből, az összeget integrállal helyettesítve, azt kapjuk, hogy a lefedetlen rész r függvényében csak mint r^{1-B} (tehát $B = 5/4$ mellett a negyedik gyök reciprokával) csökken. A gyakorlatban ez annyit jelent, hogy míg a 100 ezer leggyakoribb tő felvétele 5.6% lefedetlenséget hagy, 1 millió tőnél ez 3.2%, 10 millió tőnél 1.8%, és 100 millió tőnél (ennyit korpuszunk nem is tartalmaz) 1%, és általában a lefedetlenség egy nagyságrendnyi csökkenéséhez a kezdeti korpuszt négy nagyságrenddel kell növelni.

4. Konklúzió

A fentiekben megmutattuk, hogy a helyesírás-ellenőrzők pontosságát elsősorban lefedettségük (tehát a tőtár mérete) határozza meg. A tokenek és típusok összefüggésének jólismert törvényeire támaszkodva levezettük a bővítés várható hatását a lefedettségre, amit a méréseink alá is támasztottak. Az emberi erőforrások minimalizálása miatt tehát a lefedettség növelését egy határon túl a feldolgozandó szövegek heurisztikákkal operáló előszelekciójával (a mondatközi nagy kezdőbetűs szavak nagy valószínűséggel tulajdonnevek, amelyeket fel kell venni, a csupa számjegyből álló szövegszavak viszont telefonszámok vagy dátumok, amiket viszont automatikusan el lehet hagyni) érdemes csak megkísérelni.

Hivatkozások

1. Baayen, R. H. 1996: The effect of lexical specialisation on the growth curve of the vocabulary. *Computational Linguistics* 22, 455–480.
2. Füredi M., Kornai A., Prószéky G. 2003: A SZÓTÁR adatbázis. Kézirat.
3. Kornai A. 1992: Frequency in morphology. In: I. Kenesei (ed): Approaches to Hungarian IV 246–268.
4. Kornai A. 1999: Zipf's law outside the middle range. Proc. Sixth Meeting on Mathematics of Language, University of Central Florida, 347–356.
5. Németh L. 2003: A szószablya fejlesztés. Az V. Linux konferencián elhangzott előadás cikk változata. URL http://konf2003.linux.hu/.
6. Veenker, W. 1968: Verzeichnis der Ungarische Suffixe und Suffixkombinationen. Mitteilungen der Societas Uralo-Altaica 3, Hamburg.

Word frequency and spell-checker accuracy

Péter Halácsy[1], András Kornai[2], László Németh[1], András Rung[3], István Szakadát[1], and Viktor Trón[4]

[1] Centre of Media Research and Education, Budapest University of Technology and Economics, {halacsy,szakadat,rung}@mokk.bme.hu
[2] MetaCarta Inc., andras@kornai.com
[3] Center of Cognitive Science, Budapest University of Technology and Economics, rung@itm.bme.hu
[4] School of Informatics, University of Edinburgh, v.tron@ed.ac.uk

The Szószablya (Wordsword) project[5] aims at the creation of open source Hungarian language resources. We have already collected large (over billion words) corpora from the web, and we are in the process of distilling these into more usable word frequency lists and dictionaries. We are also developing a family of low-level analytic tools, including the Hunspell spellchecker, the Hunstem stemmer, and the Hunmorph morphological analyzer, which share the same morphological analysis core and the same base dictionary, currently containing about 80k stems and their morphological subclassification.

This paper focuses on formally characterizing the level of synergy between these two efforts: to what extent can frequency-ordered lists of word-forms be exploited for improving the quality of a stemmer, morphological analyzer, or spellchecker? We concentrate on this last case (since Hunspell is already available on SourceForge) but note here that our analysis carries over without significant changes to the pure stemming/morphological analysis task as well.

First we define the error of a spellchecker given a non-interactive scenario, when the analysis of each word can result in acceptance, rejection (with or without suggesting alternatives), or the spellchecker explicitly noting that the word is outside its scope. Under realistic assumptions about the inherent error rates of the morphological analysis component and the morphological information contained in the stemmer, it turns out that the driving factor of decreasing the error of a spellchecker is the amount we can increase its scope. Next we assume a greedy algorithm, whereby the spellchecker dictionary is gradually increased to include the stems for the first r word forms in frequency order. We use "frequency of frequencies" statistics obtained from some of our larger corpora (670m and 113m words) to demonstrate that Zipf's Law $p_r \sim 1/r^B$ offers a reasonable statistical characterization of Hungarian with $B = 1.25$. Finally, we compute the frequency of word forms left out of scope by a spellchecker based on the first r stems, and conclude that for Hungarian this decreases only with the fourth root of rank which suggests a practical limit to the corpus-sampling technique of boosting stem-dictionary coverage.

[5] managed by the Centre of Media Research and Education of Budapest University of Technology and Economics, supported by Axelero Internet and Ministry of Informatics

10-16 éves tanulók írásbeli szókincsének gyakorisági szótára[1]

Cs. Czachesz Erzsébet[1], Csirik János[2]

[1]Department of Education, University of Szeged, H6722, Szeged, Petőfi sgt. 30-34.
[2]Department of Computer Science, University of Szeged, H6720, Szeged, Árpád tér 2.

Absztrakt. Az elkészített gyakorisági szótár 4., 6., 8., 10. osztályos tanulók fogalmazásai alapján készült. A minta országosan reprezentatív, a számítógépes elemzés 2170 tanulóra terjedt ki. A teljes korpusz mintegy 600.000 szót tartalmaz.

A gyakorisági szótárak sokféle tudomány és alkalmazási terület fontos forrásai. A továbbiakban a jellegzetes fejlődési irányokat és alkalmazási területeket mutatjuk be röviden. Történetük a 19. század második felére nyúlik vissza, azóta - becslések szerint - mostanáig körülbelül húsz nyelvre mintegy félezer gyakorisági szótár készült. Angol, francia és német nyelvterületen átlagosan öt-hat évenként jelenik meg új.

A magyar nyelvre eddig kevés gyakorisági szótár született. Más szótártípusoknak, így például értelmező, történeti-etimológiai, írói életműveket feldolgozó, asszociációs, tájszótáraknak vannak értékes hagyományai.

Általánosabb célú, a köznyelv valamely rétegének szókészletét feldolgozó gyakorisági szótárunk 1941-ben jelent meg. Nemes Zoltán készítette 401 000 újságnyelvi szövegszó alapján. Ennek közvetlen előzménye az 1933-ban, ugyancsak Nemes Zoltán által jegyzett parlamenti nyelvi gyakorisági szótár. Cser János 1939-ben publikálta szótár formájában a gyermekek szókincsével kapcsolatos kutatásainak eredményeit. A mi munkánk a magyar előzmények közül leginkább ehhez kapcsolódik. 1989-ben jelent meg Füredi Mihály és Kelemen József szerkesztésében a szépprózai gyakorisági szótár, néhány évvel azelőtt pedig a jelen kötet jegyzői által publikált újságnyelvi gyakorisági szótár (Cs. Czachesz Erzsébet és Csirik János, 1986).

Nemzetközileg mérföldkőnek számít ezen a területen Gamble (1861) munkája, aki a kínai ideogrammák gyakoriságát vizsgálta, célja a könyvnyomtatás fejlesztése volt. A gyakorisági szótárak készítésének azonban, a nyelvtudományi alapkutatásokon túl, - a számítógépes szövegfeldolgozás megjelenése és tömeges igénnyé válása előtt - általában valamilyen tág értelemben vett oktatási és nevelési célja volt.

Az első és legfontosabb alkalmazási területük a nyelvoktatás. Az idegennyelv-oktatásban és az anyanyelvi nevelésben is az írás, a beszédkészség, a beszédértéskészség, és az olvasási képesség fejlesztésének tervezésekor gyakran

[1] Az előadás szövege a megjelent szótár (Cs. Czachesz és Csirik, 2002) bevezetőjének rövidített és átdolgozott változata.

fordulnak a kutatók és a tananyagkészítők a gyakorisági szótárakhoz, hogy megtudják, hogy az elsajátítandó szavak milyen gyakorisággal fordulnak elő az adott nyelvben vagy rétegnyelvben (például: West, 1935; Zeno és munkatársai, 1995).

A tizenkilencedik század végén és a huszadik század elején már készültek olyan gyakorisági szótárak, amelyeknek elsődleges célja az oktatásban való hasznosíthatóság volt. Knowles (1904) a vakok olvasókönyveinek összeállítását, Kaeding (1897) és Nemes Zoltán (1933, 1941) a hatékony gyorsírási rendszer tanítását és kialakítását kívánták segíteni. Az idegen nyelvek tanításának nagyobb hatékonysága érdekében, a tanítandó szókészlet kiválasztását először az Amerikai Egyesült Államokban végezték gyakorisági szótárakra támaszkodva. Az Eldridge (1911) által készített szójegyzéknek a célja a bevándorolt munkások nyelvtanulásának megkönnyítése volt. Ennek a szótártípusnak továbbfejlesztett változatai az úgynevezett minimumszótárak, amelyek adott nyelvek szókincséből azokat a leggyakoribb szavakat kívánják meghatározni, amelyek a nyelv valamilyen szintű elsajátításához alapvető fontosságúak (például: Horn, 1926; Bakonyi, 1930; Allwood és Wilhelmsen, 1947; Saukonnen és munkatársai, 1979).

Az utóbbi évtizedekben már többnyire reprezentatív nyelvi mintákon, számítógépes módszerekkel készített többfunkciójú gyakorisági szótárak a jellemzőek (például: Francis és Kucera, 1982; Hall és munkatársai, 1984; Peyawari, 1999; Leech és munkatársai, 2001).

Másik, még mindig közvetlenül az oktatással összefüggő alkalmazási terület az úgynevezett olvashatóság vizsgálata.

Feltételezések szerint egy szöveg olvashatóságának, így tanulhatóságának is az egyik kiemelkedően fontos jellemzője, hogy mennyi benne a ritkán használt, így az átlagos olvasó számára nagyobb eséllyel ismeretlen szó. Az olvasandó szöveg érthetőségének "megjósolására" vállalkoznak a kutatók akkor, amikor különböző olvashatósági formulák alkalmazásával kívánják a tanulásra szánt szövegek ilyen szempontú alkalmasságát vizsgálni (lásd erről részletesen: Harris és Hodges, 1995; Dale és Chall, 1987).

Gyakran használnak az olvasás és a gondolkodás folyamatainak kutatói is, kísérleteik tervezéséhez és végrehajtásához forrásként gyakorisági szótárakat. A szófelismerés mechanizmusainak kutatói például a szavak olvasás útján történő felismerése egyik lényeges befolyásolójának a szó gyakoriságát tartják. Eszerint a gyakoribb szavak felismerése általában még kontextus nélkül is rövidebb reakcióidőt igényel, mint a kevésbé gyakoriaké. Ennek az úgynevezett gyakorisági effektusnak fontos szerepe van például a mentális lexikon felépítésének kísérleti és elméleti modelljeiben is (lásd például: Forster és Chambers, 1973; Whaley, 1978).

A SZÖVEGMINTA

A tanulói szövegek egy reprezentatív fogalmazásvizsgálatból származnak. 1998-ban a József Attila Tudományegyetem (ma Szegedi Tudományegyetem) Pedagógia-Pszichológiai Intézete mellett működő MTA Képességkutató Csoport keretében, a Pedagógiai Tanszék korábbi kutatási eredményeire támaszkodva olyan vizsgálatsorozatot kezdtünk, amelynek segítségével azt kívántuk felmérni, hogy a

magyar közoktatás tanulói az ezredvégen milyen színvonalú képességekkel és készségekkel rendelkeznek. Vizsgálatainkhoz bemért, sztenderdizált mérőeszközöket vettük igénybe.

A programban felmértük az írásbeli kommunikatív képességek, így a fogalmazási képesség színvonalát is. A vizsgálatokat a 4., 6., 8., és 10. osztályosok köréből szervezett országos reprezentatív mintákon végeztük. A minták a 4.-8. évfolyamon a településtípusok, a 10. évfolyamon pedig az iskolatípusok szerint reprezentatívak.

A fogalmazásvizsgálatban minden tanuló két különböző időpontban írt a megadott műfajban és témában egy-egy fogalmazást. Mindegyik műfajban megadtuk a címet, az érvelés esetében két címet is, azon belül további választási lehetőség is volt, aszerint, hogy szeret-e a fogalmazásíró iskolába járni. Az egyik műfaja elbeszélés ("Egy érdekes napom"), a másiké érvelés ("Milyen felnőtt szeretnék lenni?"vagy pedig: "Miért (nem) szeretek iskolába járni?"). Mindegyik fogalmazás írásához 45-45 perc állt a tanulók rendelkezésére. A fogalmazásokat két, egymástól független bíráló hat szempontból (tartalom, szerkezet, stílus, helyesírás, külalak, összbenyomás) értékelte, ezeket az eredményeket használtuk fel (Molnár, Vidákovich és Cs. Czachesz, 2001) a fogalmazási képesség fejlődésének elemzéséhez. Összesen 8670 tanulói fogalmazást elemeztünk, amelyekből évfolyamonként, megyénként és iskolatípusonként külön kisorsoltunk 2170-et, az összes dolgozat negyedét, amelyeket számítógépen rögzítettünk. A rögzítés során változatlanul hagytuk az eredeti formákat, így például helyesírási hibákkal együtt gépeltük le a szöveget. Ezek a fogalmazások képezik az írásbeli tanulói nyelvhasználat szókincsvizsgálatának szövegmintáját.

A SZÖVEGFELDOLGOZÁS MÓDJA

A tanulói szövegek további feldolgozása egy másik kutatási téma és kutatóhely keretében folytatódott. A Szegedi Tudományegyetem Informatikai Tanszékcsoportja mellett működő Mesterséges Intelligencia SZTE-MTA-Kutatócsoport és a MorphoLogic Kft. egy IKTA-pályázat támogatásával, az írott szövegek szövegszavai morfológiai elemzésének és szófaji egyértelműsítésének algoritmikus lehetőségeit vizsgálja. A kutatás tágabb kontextusa az a nemzetközi és hazai informatikai és nyelvészeti kutatási irány és törekvés, amelynek hosszabb távú célja a természetes nyelvek (így a magyar is), gépi feldolgozási lehetőségeinek az előkészítése, illetve megteremtése. A gépi feldolgozás szükségességét nem csupán az Internet és használatának világméretű hódító útja, hanem a gépi (géppel segített) fordítás iránti egyre növekvő igény is indokolja.

Az IKTA-pályázat célja: a kutatók által fejlesztett, úgynevezett tanulási algoritmusok segítségével meghatározott és előre megadott szabályok szerint legyen képes egy program a szövegszavak szótári szavakká való átalakítására. Az ebben a projektben használt korpusz (szövegminta) milliós nagyságrendű, a tanulói fogalmazások összesen körülbelül 600 ezer (de külön kezelt) szövegszavából körülbelül 200 ezer szó méretű anyag is része a teljes korpusznak.

A szövegszavak szótári szavakká való átalakítása folyamán első lépésként minden szó megfelelő szófaji és morfológiai címkéket kapott. (Gépi annotáció.) Az annotáció

alapja a MorphoLogic Kft. által kifejlesztett HuMor elnevezésű magyar nyelvi elemző szoftver, valamint az európai nyelvekre kidolgozott, úgynevezett MSD kódrendszer (Alexin és munkatársai, 1999).

A feldolgozás következő lépéseként a projektben közreműködő egyetemi hallgatók rövid szövegkontextus alapján ellenőrizték, javították és kiegészítették a gépileg kapott annotációk helyességét. A szövegszavak relatív tövének és morfológiai elemzésének ellenőrzésekor a referencia — amikor ez lehetséges volt —a Magyar értelmező kéziszótár (Juhász és munkatársai, 1972) volt.

A szövegszavak szótári szavakká való alakításakor a magyar nyelv jellemzői és a nemzetközi kódolási minták alapján a következő szófaji kategóriákat vettük számításba:

Kategória*	Kód	Kategória	Kód
Melléknév és melléknévi igenév	A	Határozószó, igekötő, határozói igenév	R
Kötőszó	C	Névutó	S
Indulatszó, mondatszó	I	Névelő	T
Számnév	M	Ige	V
Főnév	N	Rövidítés	Y
Névmás	P		

**Megjegyezzük, hogy a szófajilag annotált, egyértelműsített korpusz ennél sokkal részletesebb szófaji információkat tartalmaz. Az alkalmazott MSD kódrendszerben lehetőség volt a ragozási, képzői információk tárolására is.*

A gyakorisági szótár elkészítésekor csak az általános szófaji kategóriákat vettük számításba. A tulajdonneveket nem szerepeltetjük a szótárban, de egy külön tulajdonnévi tárban tároltuk, így a további kutatásra rendelkezésre állnak. A tulajdonnevekből képzett melléknevek viszont megtalálhatóak a szótárban.

A teljes feldolgozott korpusz, amelyhez mindenféle további kutatási célból hozzá lehet férni, a következő web-címen található: http//: www. inf. u-szeged.hu/III/iskolascorpus.html.

A SZÓTÁR FELÉPÍTÉSE

Először abc-rendben adjuk meg a teljes korpusz valamennyi előfordulásra eső összes szavát. Ez a teljes lista lehetővé teszi, hogy az olvasó érdeklődésének megfelelően pótlólagos információkhoz juthasson azokban az esetekben, amikor — helytakarékossági okokból— nem teljes egészében közöljük a részmintákat és a szófaji mintákat.

Ezután felsoroljuk a teljes anyag leggyakoribb 1000 szavát, majd szófajonkénti listákat közlünk. A teljes korpusz szófaji listáin általában az első 600 gyakorisággal bezárólag szerepeltetjük a szavakat, kivéve, ha az előfordulások viszonylag alacsonyabb száma miatt a teljes felsorolást adhatjuk meg.

A szótár második részében életkoronkénti sorrendben közöljük a teljes korpuszban követett eljárás szerint a részminták adatait. Először a negyedikesek, majd a hatodikasok, a nyolcadikasok, végül pedig a tizedik osztályos tanulók fogalmazásainak mintájából a leggyakoribb szavakat (az első 500-500 szót) közöljük. Ezután mindegyik életkorban a szófaji gyakoriságokat adjuk meg.

Az utolsó részben az olvasó összefoglaló táblázatokat találhat, amelyben a teljes minta és az életkoronkénti részminták legfontosabb gyakorisági adatait foglaltuk össze.

Bibliográfia

Alexin, Z.; Váradi, T.; Oravecz, Cs.; Prószéky, G.; Csirik, J.; and Gyimóthy, T. (1999): *FGT-A Framework for Generating Rule-based Taggers.* ILP-99 Late-Breaking Papers, Bled, 24-27 June, p. 1-7. http://www.cs.bris.ac.uk/ilp99/

Allwood, S. and Wilhelmsen, I.(1947): *Basic Swedish Word List.* Rock Island.

Bakonyi, H. (1930): *Die gebrauchlisten Wörter der deutschen Sprache.* München.

Cs. Czachesz Erzsébet és Csirik János (1986): *Újságnyelvi gyakorisági szótár.* Szeged, Budapest, Debrecen; Magyar Pszicholingvisztikai Tanulmányok, IV.

Cs. Czachesz Erzsébet és Csirik János (2002): *10-16 éves tanulók írásbeli szókincsének gyakorisági szótára.* Budapest, Books in Print.

Cser János (1939): *A magyar gyermek szókincse. Gyakorisági és korszótár.* Budapest, Magyar Pedagógiai Társaság.

Dale, S. and Chall, J. S. (1987): *Readability revisited.* New York, McGraw-Hill.

Eldridge, R. C. (1911): *Six Thousand Common English Words.* Niagara Falls.

Forster, K. I. and Chambers, S. M. (1973): *Lexical Acces and naming time.* Journal of Verbal Learning and Verbal Behavior, 12, 627-635.

Francis, W. N. and Kucera, H. (1982): *Frequency Analysis of English Usage: Lexicon and Grammar.* Boston, Houghton Miffin.

Füredi Mihály és Kelemen József (1989): *A mai magyar nyelv szépprózai gyakorisági szótára.* Budapest, Akadémiai Kiadó.

Gamble, W. (1861): *Two Lists of Selected Characters Containing all in the Bible and Twenty-Seven Other Books.* Sanghai.

Hall, W. S.; Nagy, W. E. and Linn, R. (1984): *Spoken Words: Effects on Situation and Social Group on Oral Word Usage and Frequency.* Hillsdale, New Jersey, Lawrence Erlbaum.

Harris, T. L. and Hodges, R. E. (1995): *The Literacy Dictionary. The Vocabulary of Reading and Writing.* Newark, International Reading Association.

Horn, E. (1926): *A Basic Writing Vocabulary.* Iowa City.

Juhász J.; Szőke I.; O. Nagy G.; Kovalowsky m. (1972): *Magyar értelmező kéziszótár.* Budapest, Akadémiai Kiadó.

Kaeding, F. W. (1897): *Haufiigkeitswörterbuch der deutschen Sprache.* Berlin, Steglitz.

Knowles, J. (1904): *The London Print System of Reading for the Blind.* London.

Leech, G.; Rayson, P. and Wilson, A. (2001): *Word Frequencies in Written and Spoken English based on British National Corpus.* Harlow, Longman.

Molnár Edit Katalin, Vidákovich Tibor, Cs. Czachesz Erzsébet (2001): *Writing Development: The Role of School-related and Socioeconomic Factors.* Paper presented at the9th European EARLI-Conference, Switzerland, Fribourg.

Nemes Zoltán (1933): *A magyar parlamenti nyelv leggyakoribb szavai.* Szeged, Az Egységes Magyar Gyorsírás Könyvtára, 66.

Nemes Zoltán (1941): *Szóstatisztika egymillió szótagot felölelő újságszövegek alapján.* Szeged, Az Egységes Magyar Gyorsírás Könyvtára, 190.
Peyawari, A. (1999): *The Core Vocabulary of International English: A Corpus Approach.* Bergen, The Humanities Information Technologies Research Programme.
Saukonnen, P. et al. (1979): *A Frequency Dictionary of Finnish.* Perwoo-Helsinki-Juwa.
West, M. (1935): *Definition Vocabulary.* University of Toronto, Department of Educational Research, Canada.
Whaley, C. P. (1978): *Word-nonword Classification Time.* Journal of Verbal Learning and Verbal Behavior, 17, 143-154.
Zeno, S. M.; Ivens, S. H.; Millard, R. T. and Duvvuri, R. (1995): *The Educators Word Frequency Guide.* New York, TouchstoneApplied Science Associates.

Word frequency dictionary of the written vocabulary of 10- to 16-year-olds

Erzsébet Cs. Czachesz and János Csirik

[1]Department of Education, University of Szeged, H6722, Szeged, Petőfi sgt. 30-34.
[2]Department of Computer Science, University of Szeged, H6720, Szeged, Árpád tér 2.

Keywords: frequency dictionary, child language

The presentation outlines the preparatory work for a frequency vocabulary and highlights the most important findings of the project. The previous child language word frequency dictionary in Hungarian was compiled by János Cser in 1939, as an outcome of his research on children's vocabulary.

Before the appearance and subsequent widespread need for computerised text processing, the compilation of frequency vocabularies usually served educational objectives besides the purposes of basic linguistic research. The first and most important area of their application is *foreign language teaching*. In recent decades, multifunctional frequency dictionaries based on representative linguistic corpora have appeared. Another area of application, also related to education, is the analysis of *readability*. Researchers of reading and reasoning processes also often rely on frequency dictionaries as resources in developing and executing experimental designs.

In the present dictionary project, student texts were collected in a representative survey of written composition. In grades 4, 6 and 8, sub-samples are representative for settlement types and in grade 10, for school stream. Altogether 8,670 student compositions were analysed. Of these, one fourth, 2,170 were randomly selected and entered into a computerised corpus. Data processing was carried out in the SZTE-MTA Research Group on Artificial Intelligence at the Institute of Informatics of the University of Szeged. In the process of text annotation, every word was assigned the appropriate grammatical and morphological labels. The annotation was based on HuMor, a Hungarian linguistic analysis software developed by MorphoLogic Kft, and on the MSD code system developed for European languages.

The dictionary first gives an alphabetical list of all the word frequencies in the whole corpus. Next the most frequently occurring 1,000 words are given, then lists for parts of speech. In the second half of the dictionary, the same structure is followed in presenting data for age based sub-samples. Frequencies for parts of speech are given for all grades. Data are summarised in tables in the last section of the dictionary.

Idői struktúrák feltárása kvalitatív és kvantitatív szövegelemzéssel

Huszár Zsuzsanna[1], Dr. Sramó András[2]

[1]PTE BTK Tanárképző Intézet, 7624 Pécs, Ifjúság útja. 6.
huszped@tki.pte.hu,
[2]PTE IGYFK Gazdaságtudományi Intézet, 7100 Szekszárd, Rákóczi út 1.
sramo@igyfk.pte.hu

Kivonat. Az előadás megkísérli egy PhD kutatás kezdeti lépéseit felvázolni a számítógépes nyelvészet kvantitatív technikáinak és a kvalitatív tartalomelemzési adatok számítógépes feldolgozási technikáinak ötvözésével, a tartalomelemzés szempontjainak bemutatásával és a szóstatisztikai vizsgálatok néhány eredményének közzétételével a 10-16 éves korosztály iskolai fogalmazásainak 600 ezer szavas reprezentatív mintájának bázisán.

Kulcsfogalmak: szövegelemzés, szóstatisztika, Zipf-törvény, kvalitatív tartalomelemzés, idői struktúra

1 Bevezetés

A folyamatban lévő PhD-kutatás, amelyről előadásunk szól, két doktori programhoz is szoros szálakkal kötődik: egyrészt a Szegedi Tudományegyetem korábbi Neveléstudományi Doktori Programjához, s ennek Cs. Czachesz Erzsébet által irányított, a kommunikatív kompetencia fejlődésével és fejlesztésével foglakozó alprogramjához, másrészt a Pécsi Tudományegyetem Alkalmazott Nyelvészeti Doktori Iskolájához.
A kutatás szövegbázisa a 10-16 éves korosztály iskolai fogalmazásainak a Szegedi Tudományegyetem kutatásai számára rögzített, 2170 fogalmazásból álló, mintegy 600 ezer szavas, országos, reprezentatív mintája.

2. A kutatási elképzelések leírása

A kutatás az idő, ill. az idői struktúrák nyelvi megjelenésére, nyelvi létezésmódjára irányul. Vizsgálatunkban a fizikai időfogalommal szemben az idő társadalmi vetülete (tempus) kerül előtérbe. Alapkérdésünk, hogy hogyan bukkan elő, és milyen kontextusba ágyazottan az idő, illetve az idöiség a gyermek-fogalmazásokban, hogyan ragadható meg strukturáló ereje a gondolkodásra nézve. A kutatás célja az idöiség nyelvi megjelenésének komplex vizsgálata, s az elemzési célokkal adekvát

kritériumrendszer és alkalmas számítógépes szövegfeldolgozó eljárás kidolgozása, ennek segítségével egy ellenőrizhető fogalmazástipológia kialakítása, majd ezen tipológia összevetése a kutatás független változóival, végül az eredmények interpretálása a neveléstudomány és az alkalmazott nyelvészet számára.
Független változóink: a településtípus, az iskolatípus, az évfolyam, a nem valamint a fogalmazások típusa, érvelő vagy elbeszélő jellege. Függő változók a kvantitatív és kvalitatív szövegelemzés által feltárt szövegjellemzők. Jelen előadásunkban a kavntitatív technikák közül a szóstatisztikát, a kvalitatív technikák közül pedig a tartalomelemzést emeljük ki, utalva a tartalomelemzés számítógépes lehetőségeire [4] is. A tartalomelemzés hazai irodalmában az idő önálló szempontként mint a narratív pszichológia kompozíciós elve nyer kifejtést [2].
A készülő disszertáció pedagógiai szempontból is értelmezhető diagnózis lenne arról, hogy miféle idői perspektívában gondolkoznak a különböző életkorú tanulók, és arról, hogy megjelenik-e, s hogyan jelenik meg iskolai fogalmazásaikban, érvrendszerük mögöttesében a gondolkodás távlatossága. A megismeréstudomány és a nyelvészet közös metszetében elemezhető volna, hogy van-e összefüggés a nyelv absztrakciós szintje és a gondolkodás távlatossága között.

2.1 Előfeltevések

Az időnek strukturáló szerepe van. Kulturális szempontból az idői struktúrák lényegiek. A grammatikai struktúrákon túl, illetve azoktól némiképp elkülöníthetően egy szöveg idői struktúrája mint a múltra, jelenre, jövőre vonatkozó tartalmak megkülönböztetéséből és arányaiból adódó struktúra is értelmezhető és elemezhető. Az időhöz kötöttség nem feltétlenül szándékolt eleme egy szövegnek, amennyiben a szöveg idővonatkozásai, időarányai nem előre eltervezettek. Az egyes fogalmazásokon túl a szöveg-egészben megjelenik valamiféle időszemlélet, amely az idővel való társadalmi bánásmódként válik azonosíthatóvá. A szövegek idői struktúrája a fenti értelemben nem más, mint az időhöz való társadalmi viszony kifejeződése.

2.2 Alaphipotézisek

Alaphipotézisünk szerint a nyelv univerzális jellege az általunk vizsgált szövegmintában is tükröződik. Az univerzalitást mint a Zipf-törvény [5] érvényesülését vizsgáljuk előadásunkban. Feltevésünk szerint a fogalmazások idői struktúrái a társadalmi szinten szabályozott "racionális" időfelhasználást tükrözik, vagyis az időnek instrumentális jelentősége van., s az idő lineáris ill. logaritmikus szemlélete domináns. Feltételezzük továbbá, hogy az időszemlélet nyelvi megragadása világképre utaló jelentőséggel bír.

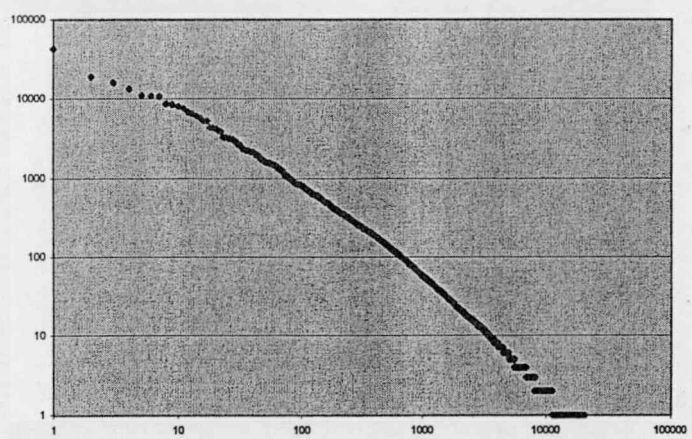
2.3 A kvalitatív elemzés tartalmi szempontjai

1. Előremutató és visszautaló idői perspektíva és ezek aránya a szövegekben.
2. A szövegben foglalt tartalmak átfogó idői tartománya.

3. A múlthoz, a jelenhez, és a jövőhöz kapcsolódó kijelentések aránya.
4. Múlthoz, jelenhez, jövőhöz kapcsolódó kijelentések értéktelítettsége: pozitív, negatív, semleges tartalma. Az átértékelés időhöz kötött mozzanatai és jellegzetes tartalmai.
5. A pozitív, negatív és semleges tartalmakhoz kapcsolódó földrajzi, topológiai kategóriák; az értéktartalmak s a térbeli viszonyok egymásra vetítése.
6. Szubjektív és objektív időészlelés. A szubjektív időélmény jellemzői. Explicit hiányérzetek: mire nincs idő?
7. A fogalmazások alapritmusát kijelölő időbeli viszonyok, napi, heti, éves stb. ritmusok megjelenési gyakorisága.
8. Az iskolához és az iskolában töltött időhöz valamint a feltételezett tanulói szerepelvárásokhoz való érzelmi viszony.
9. Az iskola funkciói a tanulók deklarációiban.
10. Az érvek forrása; írásbeliségre és szájhagyományra hivatkozó utalások aránya.
11. A személyes hivatkozási rendszer időbeli jellemzői: intragenerációs, intergenerációs, többgenerációs utalások és ezek aránya. Az eleven emlékezet terjedelme.

3. Kezdeti eredmények

Az említett korpusz önálló szóstatisztikai feldolgozását nem végeztük el, mivel rendelkezésünkre állt a korpusz gyakorisági szótára [1]. Ennek felhasználásával azt vizsgáltuk, hogy az időre vonatkozó, az időiséget kifejező, adott szófajú szavak hol helyezkednek el a szógyakoriságok által kijelölt sorrendben. Eredményeinket a Zipf-törvény segítségével mutatjuk be.

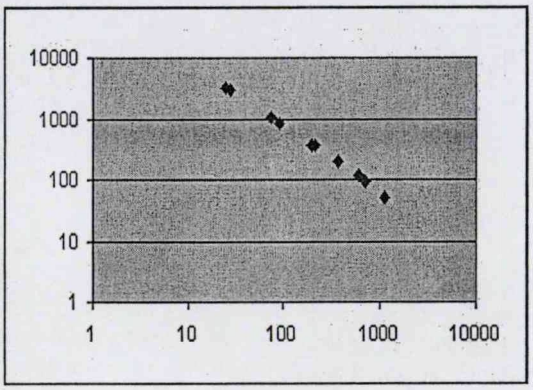


1. ábra. A teljes mintában előforduló különböző szavak gyakorisága, pozíciója a gyakorisági sorrendben (20 300 szó)

A Zipf-törvény alkalmazásával kapcsolatban meg kell említenünk, hogy a korpusz szószáma (601 135) lényegesen meghaladta az általunk ismert minták szószámát, és így egy analitikusan "szépen" viselkedő görbét kaptunk, amely a középső szakaszán jól közelíthető egy loglineáris egyenessel. A pontdiagram egyértelműen mutatja a törvény érvényesülését a vizsgált mintára, vagyis annak az univerzális szabályszerűségnek a megjelenését, amely szerint egy adott mintában az egyes szavakhoz tartozó előfordulási gyakoriságok és az ezáltal kijelölt sorrendi pozíciójuk, sorszámuk szorzata megközelítőleg stabil. Ezt a szabályszerűséget mint univerzális sajátosságot tartja számon a szakirodalom, s ennek alapján kimondható, hogy az iskolai fogalmazások is hordozzák ezen univerzális jellegzetességet.

A következőkben a teljes anyag leggyakoribb időre utaló főneveinek felsorolását adjuk meg gyakoriságuk és gyakorisági sorrendjük feltüntetésével

Sorszám	Gyakoriság	Főnév
25	3177	*nap*
28	2983	*óra*
75	1073	*idő*
93	839	*év*
206	371	*perc*
218	355	*hét*
381	200	*jövő*
612	114	*pillanat*
710	94	*hónap*
1152	49	*félév*



2. ábra. A leggyakoribb időre vonatkozó főnevek gyakorisága és előfordulási sorrendi pozíciója a teljes mintában (10 szó).

A leggyakoribb idői lépték, konkrét időtartam, amely a fogalmazásokban feltűnik, a *nap*, ezt követi az *óra*. A *nap* és az *óra* a teljes korpusz leggyakoribb főnevei között a második és a harmadik helyen szerepel, ez is megerősíti, hogy mint a fogalmazások meghatározó időegységeit tartsuk számon őket.

Az idői kifejezések gyakoriságát mint a korosztály időszemléletére utaló egyik lehetséges utalást tekintjük, s ennek alapján valószínűsítjük, hogy a korosztály gondolkodásmódjában a jelenorientáltság dominál.

A teljes mintában megfigyelt szabályszerűség az időhatározók tekintetében is érvényesül. Az időhatározók sorában kiugró gyakorisággal mutatkozik a *mindig* kifejezés (1669 említés), ami feltevésünk szerint mint állandóság és rendszeresség akár az iskolás életforma jelzője is lehet. Ezzel az uralkodó folyamatossággal szemben a végességre és az időtlenségre utaló kifejezések (*utoljára, örökre*) feltűnően kis (23-22) gyakorisággal, a többi időhatározótól erősen leszakadva jelennek csak meg. Talán mert a metafizikai dimenziókra való nyitottság nem lehet, s nem is jellemzője ennek a korosztálynak.

4. Egy konkrét példa

Bár jelen írás keretei között a tartalomelemzés osztályozási rendszerének, s a kialakított kategóriáknak és kódolási technikának a bemutatására nem vállalkozunk, előrebocsátjuk, hogy elemzésünk során lépcsőzetes osztályozási rendszert használunk, és Viekko Pietila [6] nyomán tárgyi és viszonyulási tartalmakat különböztetünk meg. Végezetül egy konkét fogalmazás szöveges jellemzését kíséreljük meg.

"Szeretek iskolába járni, mert sok mindent meg lehet tanulni a felnőttektől, amit magamtól nem tudnék megérteni." (Fogalmazáskód: 8552107418)

A fogalmazás egyetlen összetett mondat, három tagmondatból áll. Az elemzés alapjául az ebben foglalt négy kijelentési egység szolgál. A felütés pozitív, és miután a szöveg egyetlen mondat, a felütés gyakorlatilag a konklúzióval azonos. Az iskolában megélt idő értékelése pozitív, deklaráltan csak pozitívum jelenik meg, viszont a megoldás terjedelme, a feladathoz való közömbös esteleg negatív viszonyra utal. Az érvelés "minimalista", a feladat formális teljesítése a felkínált tanulószerep hárítására utal. Az érvelésmód racionális, a szerkezet logikai, ok-okozati viszonyt mutat. A szöveg tartalmi hagsúlya a tanulásra esik. Az iskola funkciójára vonatkozóan kognitív szempontok hangsúlyosak; a tanulás, a tudásátadás és a megértés. A szövegnek nincs kitüntetett időbeli kezdőpontja, sem pedig kitüntetett végpontja, átfogó idői tartománya nem explicit, hanem az iskola intézményéhez való egyéni kötődés időszakaként értelmezhető, valószínűsíthető. A fogalmazás meghatározó ritmusát az iskolába járás (napi) szokása és a felnőtt-gyerek viszony „generációs képlete" adja. Idői perspektíváját tekintve inkább jelenorientált, de utal a közeljövőre is.

Hasonlóképpen leírhatóak a többi fogalmazás jellemzői is. Ezen jellemzők kódolása, digitalizálása, táblázatba rendezése lehetővé teszi a kvalitatív elemzéssel kapott adatok kvantitaív vizsgálatát. Egy ellenőrizhető fogalmazástipológia kialakításához a továbbiakban clusteranalízis és a faktoranalízis kínálkozik célravezető eljárásnak.

Hivatkozások

1. Cs. Czachesz Erzsébet – Csirik János: 10-16 éves tanulók írásbeli szókincsének gyakorisági szótára. BIP, 2002.
2. Ehmann Bea: A szöveg mélyén. A pszichológiai tartalomelemzés. Új Mandátum, Budapest, 2002.
3. Kornai, András: Zipf's law outside the middle range. In: Proc. Sixth Meeting on Mathematics of Language. University of Central Florida, 1999. 347-356. o.
4. Krippendorff, Klaus: A tartalomelemzés módszertanának alapjai. Balassi Kiadó, Budapest, 1995.
5. Manning, Christopher D. – Schütze, Hinrich: Foundations of Statistical Natural Language Processing. The MIT press, Cambridge, 1999.
6. Pietila, Veikko: Tartalomelemzés. Tömegkommunikációs Kutatóközpont, Budapest, 1979.
7. Powers, David M. W.: Applications and Explanations of Zipf's Law. In D. M. W. Powers (ed.) NeMLaP3/CoNLL98: New Methods in Language processing and Computational Natural Language Learning, ACL, 1998. 151-160. o.

Exploration of temporal structures by qualitative and quantitave text analysis

Zsuzsanna Huszár[1], Dr. András Sramó[2]

[1]PTE BTK Teacher Training Institute, 7624 Pécs, Ifjúság útja. 6.
huszped@tki.pte.hu,
[2]PTE IGYFK Institute of Economics, 7100 Szekszárd, Rákóczi út 1.
sramo@igyfk.pte.hu

Abstract. The topic we will address in this lecture/study is the analysis of time and temporal structures by the means of computational linguistics. Our research is based on the analysis of 2170 composition (600,000 words) written by students between the ages of 10-16, who were selected from different parts of the country. The data collection was initiated and carried out by University of Szeged. Our study is connected both to the Doctoral Program in Education at the University of Szeged and the Applied Linguistics Program at the University of Pecs. The goals of this study are to analyze the temporal structures in the collected texts and through this describe a specific age group's attitude to time, as well as to create a verifiable classification system based on the collected material and compare it with the independent variables of the study. Our independent variables are: type of settlement, type of school, class, gender, genre of writing. The dependent variables are different text attributes discovered through quantitative and qualitative analysis. We applied word counts as part of the quantitative analysis and content analysis as part of the qualitative analysis. By our assumptions in the school compositions there is appearing the universal attribute of the language, and the description of the temporal structure of texts makes possible to show the attitudes to time. This is important from social and pedagogical point of view. Using word count we proved the validity of the Zipf's law on the investigated corpus. In the course of the qualitative analysis of temporal structures we described a special classification as a method for content analysis.

Keywords: text analysis, word count, Zipf's law, quantitative content analysis, temporal structures

Magyar nyelvű szótárak tömör reprezentációja nemdeterminisztikus automatákkal

Kertész-Farkas Attila[1], Fülöp Zoltán[2], Kocsor András[3]

[1,3] Szegedi Tudományegyetem, MTA-SZTE Mesterséges Intelligencia Kutatócsoport, Aradi vértanúk tere 1, 6720 Szeged *
[2]Szegedi Tudományegyetem, Számítástudomány Alapjai Tanszék, Árpád tér 2, 6720 Szeged
[1]kfa@rgai.inf.u-szeged.hu
{[2]fulop, [3]kocsor}@inf.u-szeged.hu

Kulcsszavak: automata, minimális automata, automata-tömörítés

1. Bevezetés

Azokban a számítógépes alkalmazásokban, amelyek véges nyelvekkel, nyelvi korpuszokkal dolgoznak, – mint például a beszédfelismerésben, beszédszintézisben alkalmazott programok, – a nyelvek reprezentálására hatékonyságuk és egyszerű szerkezetük miatt automatákat érdemes alkalmazni. Egy nyelv felismerésére számos egymással ekvivalens automata megkonstruálható, melyek közül a lehető legkisebb méretűt, a legtömörebbet érdemes használni.

Ebben a cikkben véges nyelveket felismerő automatákat tömörítő heurisztikus algoritmusokkal foglalkozunk. Automata tömörítő algoritmuson olyan algoritmust értünk, amely egy (általában nemdeterminisztikus) $\mathcal{A}$ automatából kiindulva megkonstruál egy $\mathcal{A}$-val ekvivalens nem feltétlenül determinisztikus, de kisebb méretű automatát.

A minimális determinisztikus automata (MDFA) megkonstruálásával sokan foglalkoztak, lásd a [W94] összefoglaló munkát, viszont nemdeterminisztikus automaták (NFA) tömörítésére eddig kevesebb figyelem esett. Közismert, hogy megadható olyan k állapotú NFA, mellyel ekvivalens MDFA-nak 2^k állapota van [AJV99]. Ugyanakkor, a minimális állapotszámú NFA kiszámolása NP-nehéz probléma [JR93], ezért heurisztikus algoritmusok kidolgozására van szükség. [AJV99]-ben egy olyan heurisztikus tömörítő algoritmust dolgoztak ki, amely egyenlő hosszúságú szavakat tároló automatán működik, és az automata gráfján definiált biklikk lefedő rendszerek alapján végzi el a megfelelő tömörítést. Ezt a módszert általánosították tetszőleges nyelvet felismerő automatákra [CC03]-ban. Ez utóbbi heurisztikus algoritmus azonban nagy időigénye miatt a gyakorlatban nem igazán alkalmazható.

A DFA-t minimalizáló algoritmusok nemcsak az állapotszám, hanem az átmenetek száma szerint is az MDFA-t konstruálják meg. A NFA-k esetében azonban

* Ez a cikk az Oktatási Minisztérium 2001/055 számú IKTA pályázata támogatásával készült.

más a helyzet. Ha csak állapotszám szerint tömörítünk, mint például az [AJV99]-ben szereplő algoritmus is, akkor az átmenetek száma akár meg is sokszorozódhat és ezáltal az automata mérete növekszik. Ezt a tényt az eddigi tömörítő algoritmusok nem vették figyelembe, így például az [AJV99]-ben megadott algoritmus a kiindulási automatához képest több mint kétszer akkora helyen tárolható automatát is adott eredményül.

A cikkben, továbbfejlesztjük az [AJV99]-ben megadott tömörítési eljárást és megadunk egy olyan gyors heurisztikus algoritmust, amely nemcsak az állapotok száma szerint tömörít eredményesen, hanem korlátozza az átmenetek számának növekedését is. Így az eljárás minden esetben garantáltan kisebb méretű automatát eredményez.

A cikk felépítése a következő. A második fejezetben definiáljuk a szükséges fogalmakat, majd a harmadik fejezetben ismertetjük az említett automata tömörítő algoritmust. A negyedik fejezetben bemutatjuk az algoritmus futási eredményeit magyar nyelvű korpuszokon, végül az ötödik fejezetben összegezzük a cikk eredményeit.

2. Definíciók

Egy H halmaz számosságát $|H|$-val jelöljük. Egy Σ ábécé feletti szavak halmazát Σ^*-gal jelöljük, Σ^* tetszőleges L részhalmazát pedig Σ *feletti nyelvnek*, vagy röviden csak *nyelvnek* nevezzük. *Nemdeterminisuztikus automatának* (NDFA-nak) nevezünk egy $\mathcal{A} = (Q, \Sigma, \delta, I, F)$ rendszert, ahol Q az állapotok véges, nemüres halmaza, Σ az input ábécé, $I \subseteq Q$ a kezdő-, $F \subseteq Q$ a végállapotok nemüres halmaza és $\delta : Q \times \Sigma \to \mathcal{P}(Q)$ az átmenetfüggvény. Ha I egy elemű és minden $a \in \Sigma$-ra és $q \in Q$-ra $\delta(q, a)$ legfeljebb egy elemű, akkor $\mathcal{A}$ *determinisztikus* (röviden DFA). Tetszőleges $q \in Q$-ra és $E \subseteq Q$-ra legyen $\gamma^+(q) = \{(a, q') \in \Sigma \times Q \mid q' \in \delta(q, a)\}$, $\gamma^+(E) = \bigcup_{q \in E} \gamma^+(q)$ és $\gamma^-(q) = \{(q', a) \in Q \times \Sigma \mid q \in \delta(q', a)\}$, továbbá $q^+ = \bigcup_{a \in \Sigma} \delta(q, a)$ és $q^- = \{q' \in Q \mid (\exists a \in \Sigma)\, q \in \delta(q', a)\}$.

A δ függvényt kiterjesztjük $\mathcal{P}(Q) \times \Sigma^* \to \mathcal{P}(Q)$ típusú leképezéssé úgy, hogy minden $w \in \Sigma^*$-ra és $a \in \Sigma$-ra $\delta(q, wa) = \delta(\delta(q, w), a)$, majd $\delta(E, w) = \bigcup_{q \in E} \delta(q, w)$. Az $\mathcal{A}$ automata által felismert nyelven az $L(\mathcal{A}) = \{w \in \Sigma^* \mid \delta(I, w) \cap F \neq \emptyset\}$ nyelvet értjük.

Azt mondjuk, hogy $\mathcal{A}$ *egyértelmű* (röviden UFA), ha minden $w \in L(\mathcal{A})$ szóra $\mathcal{A}$ gráfjában pontosan egy út vezet valamely I-beli kezdőállapotból valamely F-beli végállapotba. Minden DFA egyben UFA is. Az $\mathcal{A}$ automata transzponáltján az $\overleftarrow{\mathcal{A}} = (Q, \Sigma, \delta', F, I)$ automatát értjük, ahol $\delta'(q, a) = \{p \mid q \in \delta(p, a)\}$. A cikkben csak olyan automatákkal foglalkozunk, amelyeknek a gráfja nem tartalmaz kört, tehát amelyek véges nyelveket tárolnak, ismernek fel. Feltesszük továbbá, hogy egy automata minden állapota elérhető valamely kezdőállapotból és minden állapotból eljuthatunk valamely végállapotba. Szükség esetén további részletek [AJV99]-ben találhatók.

3. Automata tömörítés

Ebben a fejezetben az $\mathcal{A}$ automatán az $\mathcal{A} = (Q, \Sigma, \delta, I, F)$ automatát értjük.

Definíció 1. Legyen $q \in Q$ és $S \subseteq Q$. A (q, S) párt *$\mathcal{A}$-beli egyesítéses tömörítésnek* nevezzük, ha $q \notin S$, $\gamma^+(q) = \gamma^+(S)$, a $\{q\} \cup S$ halmaz mindegyik eleme végállapot vagy egyike sem az, és teljesül, hogy ha $q \in I$ akkor $S \subseteq I$.

A (q, S) egyesítéses tömörítés $\mathcal{A}$-ra való alkalmazása azt jelenti, hogy a q-ba érkező átmeneteket átirányítjuk minden $p \in S$ állapotba, majd q-t és a belőle induló átmeneteket töröljük. Minden egyesítéses tömörítés elvégzése után az automatában az állapotok száma eggyel csökken.

A (q, S) egyesítéses tömörítés *diszjunkt*, ha minden különböző $s, s' \in S$-re $\gamma^+(s) \cap \gamma^+(s') = \emptyset$ és $\{q\} \cup S$ egyik eleme sem végállapot. Ha $\mathcal{A}$ UFA és egy (q, S) diszjunkt egyesítéses tömörítést alkalmazunk rá, akkor a kapott automata ugyancsak UFA lesz. Természetesen az $\mathcal{A}$-beli diszjunkt egyesítéses tömörítések száma általában kevesebb mint az egyesítéses tömörítések száma.

Ha a q-ba érkező átmenetek száma olyan nagy, hogy a tömörítés elvégzése után több élt kapunk, mint amennyit sikerül megspórolni, akkor a tömörítés növeli az automata méretét. Ennek kezelésére vezetjük be a következő fogalmat, mely cikkünk egyik legfontosabb eleme. A (q, S) egyesítéses tömörítés *valódi*, ha teljesül, hogy

$$|q^-| * (|S| - 1) < |q^+| + 1$$

Látható, hogy az egyenlőtlenség bal oldalán a tömörítés után kapott új átmenetek száma áll, míg a jobb oldal mutatja azt, hogy a tömörítéssel mennyi átmenettel és állapottal lesz kisebb az automata mérete. Tehát egy egyesítéses tömörítés akkor tömörít valóban, ha a fenti egyenlőtlenség teljesül.

Egyesítéses tömörítések egy $T = \{(q_1, S_1), (q_2, S_2), \ldots, (q_l, S_l)\}$ halmazát *megengedettnek* hívjuk, ha minden $1 \leq i \leq l$-re teljesül $q_i \notin \bigcup_{i=1}^{l} S_i$.

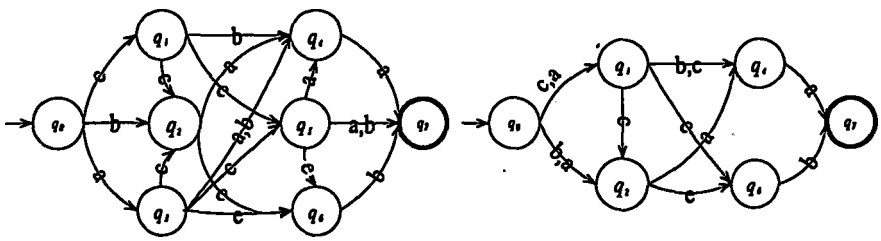
Definíció 2. Az $\mathcal{A}$ automatából egyesítéses tömörítések egy megengedett $T = \{(q_1, S_1), (q_2, S_2), \ldots, (q_l, S_l)\}$ halmazával kapott automatán azt az $\mathcal{A}' = (Q', \Sigma, \delta', I', F')$ automatát értjük, melyre

- $Q' = Q \setminus \{q_1, q_2, \ldots, q_l\}$, $I' = I \cap Q'$, $F' = F \cap Q'$,
- $\forall\, q \in Q'$ és $a \in \Sigma$ esetén $\delta'(q, a) = (\delta(q, a) \setminus \{q_{i_1}, \ldots, q_{i_k}\}) \cup S_{i_1}, \cup \cdots \cup S_{i_k}$, ahol $\{i_1, \ldots, i_k\} = \{1 \leq i \leq l \mid (q, a) \in \gamma^-(q_i)\}$.

A fenti definícióban a T egyesítéses tömörítésre vonatkozó megengedett feltétel szükséges ahhoz, hogy $\mathcal{A}'$-t egyértelműen definiálni tudjuk. A továbbiakban egyesítéses tömörítések halmazán mindig megengedett halmazt értünk, ezért a megengedett jelzőt el is hagyjuk.

Az 1. ábrán egy példa látható egyesítéses tömörítések egy halmazának alkalmazására, ahol a jobb oldali automatát a bal oldali automatából a $T = \{(q_3, \{q_1, q_2\}), (q_5, \{q_2, q_4, q_6\})\}$ halmazzal kaptuk, és így 66%-os tömörítést sikerült elérni.

Egy $\mathcal{A}$ NFA és a belőle tömörítéssel kapott $\mathcal{A}'$ NFA között a következő kapcsolat állapítható meg.



1. ábra. Példa egyesítéses tömörítések alkalmazására

Tétel 1. *Ha $\mathcal{A}'$ automatát az $\mathcal{A}$ automatából egyesítéses tömörítések egy T halmazával kapjuk, akkor $L(\mathcal{A}) = L(\mathcal{A}')$. Továbbá, ha $\mathcal{A}$ UFA és T elemei diszjunktak, akkor $\mathcal{A}'$ is UFA.*

Bizonyítás. A bizonyítás az [AJV99]-ben található hasonló állítás (Propozíció 2) bizonyításának általánosításával végezhető el.

Látható, hogy ha T valódi egyesítéses tömörítések halmaza, akkor a kapott $\mathcal{A}'$ automata állapot és átmenet száma kisebb, mint a kiindulási $\mathcal{A}$ automatáé, és így kisebb helyen tárolható.

Most megadjuk az általunk javasolt heurisztikus automata tömörítő algoritmust. Az algoritmus annyival több a [AJV99]-ben szereplőtől, hogy nemcsak azonos hosszúságú szavakat felismerő automatákat tud tömöríteni és, hogy csak valódi tömörítéseket enged meg.

3. Algoritmus. *ReductionAutomata*
input: *$\mathcal{A}$ automata*
output: *a kiindulási automatával ekvivalens, tömörített automata*

1 $n \leftarrow |Q|$;
2 *ismételjük meg kétszer:*
3 *keressük meg az összes valódi diszjunkt egyesítéses tömörítést, majd végezzük el a tömörítéseket a 2. definíció szerint;*
4 $\mathcal{A} \leftarrow \overleftarrow{\mathcal{A}}$;
5 **ha** $n \neq |Q|$ **akkor** *ugorjunk 1-re;*
6 **return** $\mathcal{A}$;

Ha az algoritmus 3. sorában nemcsak diszjunkt egyesítéses tömörítéseket keresünk, akkor az eredményül kapott automata még kisebb lesz, viszont az így módosított algoritmus nem őrzi meg az UFA tulajdonságot.

Az algoritmus úgy gyorsítható fel jelentősen, hogy egy (q, S) egyesítéses tömörítés meghatározása esetén a q állapothoz az S halmazba nem az egész automatában keresünk állapotokat, hanem csak a q állapotból elérhető állapotok őseit vizsgáljuk. Így egy, a gyakorlatban gyors algoritmust kapunk.

4. Futási eredmények

A teszteléshez néhány internetes portál szavaiból készítettünk adatokat. A tömörítő algoritmust egy Pentium III 700 MHz-es számítógepen futtatuk. A kiindulási automaták konstruálása úgy történt, hogy először standard algoritmussal egy fa gráfú DFA-t készítettünk, majd alkalmaztuk az eddig ismert leggyorsabb minimalizáló algoritmust [H71]. Ezután az így kapott MDFA-ra alkalmaztuk a cikkben javasolt automata tömörítő algoritmust. Négy automatát vizsgáltunk, a futási eredmények az 1. táblázatban láthatók.

szavak száma	kiindulási MDFA átmenetek és állapotok száma	futási idő (s)	tömörített automata átmenetek és állapotok száma	futási idő (s)	tömörítés aránya
8213	21432	2	20356	1	94.98%
112584	241463	1517	222961	226	92.34%
26852	63561	82	58618	15	92.22%
433367	794245	22947	738129	3517	92.93%

1. táblázat. A valódi diszjunkt egyesítéses tömörítések alkalmazásával kapott eredmények

A nyelvi korpuszok vizsgálatában gyakran előfordul, hogy további adatok – például súlyok vagy kimeneti szimbólumok – tárolására is szükség van. Ha tömören szeretnénk az automatát reprezentálni [K99] vagy ha az átmeneteket költségesen tudjuk tárolni, akkor érdemes korlátozni az átmenetek számának növekedését. A csak átmenetszám szerinti tömörítés esetén olyan esetek is adódhatnak, amikor egy új állapot akalmas hozzáadásával csökkenteni lehet az automata méretén. Ez újabb kutatási irányt vethet fel. Viszont, ha az automata állapotaihoz súlyokat rendelünk, vagy az átmeneteket alacsony költséggel tudjuk tárolni, akkor elvégezhetjük a nem valódi egyesítéses tömörítéseket is. Ha olyan rendszert implementálunk, amiben nem fontos, hogy az automata többször fogad el egy szót, vagyis nem UFA, akkor megengedhetjük a nem diszjunkt egyesítéses tömörítéseket is, és így tovább csökkenthetjük az állapotok számat. Ezért az algoritmust nemcsak valódi és diszjunkt egyesítéses tömörítések keresésére is lefuttattuk.

szavak száma	kiindulási MDFA állapotok száma	futási idő (s)	tömörített automata állapotok száma	futási idő (s)	tömörítés aránya
8213	7870	2	7235	1	91.93%
112584	81718	1517	65785	706	80.50%
26852	21995	82	18122	36	82.39%
433367	254140	22947	199045	8574	78.32%

2. táblázat. A nem csak valódi egyesítéses tömörítések alkalmazásával kapott eredmények

5. Konklúzió

A cikkben továbbfejlesztjük az [AJV99]-ben megadott tömörítési algoritmust és bemutattunk egy véges nyelveket felismerő, nemdeterminisztikus automatákra alkalmazható gyors heurisztikus tömörítő algoritmust. A tömörítés lényege, hogy az automatában (q, S) alakú egyesítéses tömörítéseket keresünk, ahol q egy állapot, S pedig állapotok egy halmaza, majd a q-ba érkező átmeneteket átirányítjuk az S-beli állapotokba és törljük q-t. Csak valódi egyesítéses tömörítéseket engedünk meg, ami biztosítja azt, hogy az automata mérete, amit az állapotok és az átmenetek együttes száma határoz meg, valóban kisebb lesz. Az algoritmust implementáltuk, majd alkalmaztuk négy automatára. Az algoritmus gyakorlati alkalmazhatóságát támasztja alá, hogy az elvégzett összehasonlító tesztek alapján a minimális determinisztikus automatánál 15-25%-kal kisebb (nemdeterminisztikus) automatát konstruál meg. Ezért érdemes lehet az algoritmust továbbfejleszteni, általánosítani tetszőleges reguláris nyelvet felismerő automtákra vagy megvizsgálni az átmenetek száma szerinti tömörítés lehetőségeit.

Hivatkozások

[AJV99] J. Amilhastre, P. Janssen and M. C. Vilarem, FA Minimization Heuristics for a Class of Finite Languages, *In: Proc. of WIA 99, Lecture Notes in Computer Science (Szerk. O. Boldt és H. Jürgensen), Vol. 2214, pp. 1-13, Springer-Verlag, 2001.*

[CC03] J.-M. Champarnaud, F. Coulon, NFA Reduction Algorithms by Means of Regular Inequalities, *In: Proc. of DLT 03, Lecture Notes in Computer Science (Szerk. Ésik Z. és Fülöp Z.), Vol. 2710, pp. 194-205, Springer-Verlag, 2003.*

[H71] Hocproft, J.E, An $n * log(n)$ algorithm for minimizing states in a finite automaton, *In: Theory of Machines and Comutations (Szerk. Y. Kohavi and A. Paz), Academic Press, New York, 1971, pages 189-196*

[JR93] Tao Jiang, B. Ravikumar, Minimal NFA problems are hard, *SIAM Journal of Computation, 22: 1117-1141, 1993.*

[K99] George Anton Kiraz, Compressed Storage of Sparse Finite-State Transducers, *In: Proc. of WIA 99, Lecture Notes in Computer Science (Szerk. O. Boldt és H. Jürgensen), Vol. 2214, pp. 109-122, Springer-Verlag, 2001.*

[W94] Bruce W. Watson, A taxonomy of finite automata minimization algorithms, *http://www.cs.up.ac.za/~watson/publications.html, 1994*

Compact representation of Hungarian vocabulary with nondeterministic finite automata

Attila Kertész-Farkas[1], Zoltán Fülöp[2], András Kocsor[3]

[1,3] University of Szeged, MTA-SZTE Research Group on Artificial Intelligence, Aradi vértanúk tere 1, 6720 H-Szeged *
[2]University of Szeged, Department of Computer Science, Árpád tér 2, 6720 H-Szeged
[1]kfa@rgai.inf.u-szeged.hu
{[2]fulop, [3]kocsor}@inf.u-szeged.hu

Keywords. finite automata, minimization, finite automata reduction

In linguistic applications that process finite languages, e.g., speech recognition and speech synthesis, it is worth to use finite automata due to their simple structure and efficiency. Several automata can be created that accept a given language, however it is important to find the possible smallest of them.

In this paper we deal with automata reduction algorithms. Broadly speaking, an automata reduction algorithm is an algorithm that, for a given nondeterministic automaton $\mathcal{A}$ accepting a finite language, delivers a smaller (nondeterministic) one that is equivalent with $\mathcal{A}$.

For a deterministin automaton $\mathcal{A}$, we can easily find the unique deterministic automaton $\mathcal{A}'$ which is minimal and equivalent with $\mathcal{A}$. However, the number of the states of a small nondeterministic automaton equivalent with $\mathcal{A}$ may be exponentially smaller than that of $\mathcal{A}'$. Therefore it is worth looking for reduction algorithms also for nondeterministic automata, even if computing the minimal nondeterministic automaton is known to be an NP-hard problem.

Another point is that the size of an automaton depends not only on the number of its states but also on the number of its transitions. Therefore, for the optimal storage, the automata reduction algorithms must also take care of the number of transitions. We note that, to our best knowledge, the conventional automata reduction algorithms do not consider the number of transitions, and thus in some cases they can even increase the size of the automaton.

In this paper we present a novel heuristic automata reduction algorithm. The algorithm works for arbitrary nondeterministic automata accepting a finite language and it considers both the number of transitions and states during the reduction. In the experiments we found a gain of 20-25% in the numbers of states and transitions.

* This work was supported under the contract IKTA No. 2001/055 from the Hungarian Ministry of Education.

Kézzel annotált magyar nyelvi korpusz: a Szeged Korpusz

Csendes Dóra[1], Hatvani Csaba[1],
Alexin Zoltán[1], Csirik János[1], Gyimóthy Tibor[1], Prószéky Gábor[2], Váradi Tamás[3]

[1] Szegedi Tudományegyetem Informatikai Tanszékcsoport
6720 Szeged, Árpád tér 2.
{dcsendes, hacso, alexin, csirik, gyimi}@inf.u-szeged.hu
http://www.inf.u-szeged.hu
[2] MorphoLogic Kft Budapest
1118 Budapest, Késmárki u. 8.
proszeky@morphologic.hu
http://www.morphologic.hu
[3] MTA Nyelvtudományi Intézet
1068 Budapest, Benczúr u. 33.
varadi@nytud.hu
http://www.nytud.hu

Absztrakt: A Szeged Korpusz jelenlegi állapotában egy 1.2 millió szövegszóból álló szófajilag egyértelműsített, szintaktikai szempontból laposan elemzett adatbázis. Az elemzések szabályok alapján történő automatikus előelemzést követően kézi ellenőrzéssel és javítással történtek. A folyó munkálatok keretében egy bővebb szintaktikai elemzés, azaz egy magyar nyelvű treebank építése a cél, amelyben már szemantikai információk is szerepelni fognak. A korpusz regisztráció után hozzáférhető[1], oktatási és kutatási célokra ingyenesen letölthető.

1. Bevezetés

A Szeged Korpusz jelenlegi állapota három projekt eredményeként született. Az első projekt (IKTA 2000)[2] keretében a Szegedi Tudományegyetem Informatikai Tanszékcsoportja a MorphoLogic Kft.-vel közösen egy 1 millió szövegszavas korpusz morfológiai elemzését és szófaji egyértelműsítését vállalta. A második projektben (NKFP 2001)[3] a konzorcium a Magyar Tudományos Akadémia Nyelvtudományi Intézetével kibővülve egy 200 ezer szavas szövegállomány feldolgozására, és az ez alapján történő automatikus információkinyerésre vállalkozott. Mivel a feldolgozás célja ezúttal

[1] A Szeged Korpusz a http://www.inf.u-szeged.hu/lll oldalról tölthető le.

[2] A projekt az Oktatási Minisztérium által támogatott, *Magyar nyelvi szófaji egyértelműsítő módszer fejlesztése gépi tanuló algoritmusok felhasználásával* című IKTA 27/2000. pályázat keretében valósult meg.

[3] A projekt az Oktatási Minisztérium által támogatott, *Automatikus információszerzés rövid (politikai, üzleti, piaci) hírekből* című NKFP 2/017/2001 pályázat keretében valósult meg.

irányított információszerzés volt, ezért nem csupán morfológiai elemzést foglalt magába, hanem lapos szintaktikai elemzést (főnévi csoportok, ill. tagmondatok bejelölését), és bizonyos szintű szemantikai elemzést (a főnevek és melléknevek szemantikai attribútumainak megadását, igék és vonzatkereteik közötti szemantikai összefüggések azonosítását) is tartalmazott. A harmadik, jelenleg is folyó projekt (IKTA 2002)[4] keretében a MorphoLogic Kft.-vel és a Nyelvtudományi Intézettel karöltve az előző projektekben összegyűjtött, 1,2 millió szövegszavas korpusz bővebb szintaktikai feldolgozása, azaz egy treebank építése a cél.

A Szegedi Tudományegyetem Informatikai Tanszékcsoportjának egyik mesterséges intelligenciával foglalkozó kutatói csoportja 1998-ban kezdte meg munkáját a természetes nyelvi feldolgozás területén a fent említett projektek konzorciumi partnereként. A csoport a korpusz szövegeinek összegyűjtésében és az elemzett szövegek alapján történő gépi tanulási módszerek kialakításában vállalt aktív szerepet. Mivel a szövegek kézi annotálása teljes egészében e csoport irányítása alatt zajlott, ezért kapta az adatbázis a Szeged Korpusz nevet. A nyelvi feldolgozással kapcsolatos feladatokat a csoport menedzsmentjén kívül egy 4-10 fős programozói csapat és kb. 20 nyelvészhallgató végzi. A korpuszon folyó tevékenységek két fő irányba mutatnak: egyrészt a nyelvfeldolgozás egymásra épülő szintjeinek kialakítására és annotálására, másrészt a már elkészült szintek minőségének javítására.

2. A Szeged Korpusz szöveganyaga

A korpusz szöveganyagának összeválogatásánál a fő szempont az volt, hogy a korpusz tematikailag minél sokszínűbb legyen, azaz a mai magyar nyelvnek több, egymástól különböző rétegét képviselje. Ennek megfelelően a Szeged Korpuszt hat, egymástól eltérő témájú szövegcsoport alkotja. Az első öt csoport az említett IKTA 2000 projekt keretében került összeválogatásra a szövegszavak morfológiai elemzése és szófaji egyértelműsítése céljából. A hatodik csoport a későbbi NKFP 2001 projektben, kiegészítésként került a meglévő szövegekhez. Ez utóbbi szövegcsoport szolgált az automatikus információszerzés alapjául. A vállalt projektmunkák, és a szövegeken ezen felül elvégzett elemzési munka eredményeként ma egy megközelítőleg 1,2 millió szövegszó + 225 ezer írásjel nagyságú, morfológiailag annotált, szintaktikai szempontból laposan elemzett korpusz áll rendelkezésre.

Természetesen az említett terjedelem még így sem teszi lehetővé egy átfogó, a teljes írott nyelvet felölelő szöveganyag összegyűjtését, de a konzorcium törekedett arra, hogy a korpusz a lehető legreprezentatívabb legyen, illeszkedjen az Interneten található nagy mennyiségű szöveg témáihoz. A fentieket figyelembe véve az alább részletezett témakörökből kerültek ki szövegek, egyenként kb. 200-200 ezer szavas terjedelemben. [1]

---

[4] A projekt az Oktatási Minisztérium által támogatott, *Mondatszintaxis gépi tanulása (gépi tanulási módszerek a magyar nyelv szintaktikai szabályainak létrehozására)* című IKTA 37/2002. pályázat keretében valósul meg.

Szövegtípus	Méret				Többértelmű szavak aránya szövegtípusonként	Szavak száma témakörönként
	100 mondatos fájlok száma	Szavak	Írásjelek	Többértelmű szavak		
Iskolás 1.	87	104818	22329	62705		
Iskolás 2.	73	97786	20705	52837		
Iskolás 3.	71	20454	4174	12279	57.30%	223058
Szépirod. Utas	52	60202	15946	33719		
Szépirod. Piszkos Fred	63	46578	14391	24284		
Szépirod 1984	65	80411	17631	42965	53.94%	187191
Szám.tech. Comp.World	62	124043	21018	57159		
Szám.tech. Win2000	31	57937	10888	25539	45.44%	181980
Újság (nsz)	41	63966	11980	31463		
Újság (nv)	13	22630	3900	11244		
Újság (hvg)	23	57647	9810	27990		
Újság (mh)	24	43091	7258	20678	48.78%	187334
Jogi (gazd.)	57	134872	22908	64165		
Jogi (szerz.)	35	87314	15807	42416	47.97%	222186
Rövid MTI	96	188345	25817	82813	43,7%	188345
Összesen:	**793**	**1190094**	**224562**	**592256**	**49.52%**	**1190094**

1. A korpusz összefoglaló adatai

1. Általános iskolások fogalmazásai

Ebben a témakörben 8. és 10. osztályos tanulók fogalmazásai kerültek feldolgozásra. A fogalmazások az ország különböző részein (Magyarországot egyenletesen lefedve), két különböző témában („Egy szép napom", ill. „Miért (nem) szeretek iskolába járni?" címmel) íródtak. Az előbbiből 347 elbeszélő, az utóbbiból 492 érvelő dolgozatot tartalmaz a korpusz.

2. Szépirodalmi szövegek

Irodalmi alkotások közül három regény került feldolgozásra. Szerb Antal: *Utas és holdvilág*, George Orwell: *1984*, és Rejtő Jenő: *Piszkos Fred, a kapitány* című művek teljes szövege bekerült a korpusz szöveganyagába. (Az Orwell-korpuszban[5] korábban

[5] Az összesen 98.042 szövegszót, ill. írásjelet tartalmazó, ún. Orwell-korpuszt az 1995-97 között folyó MULTEXT-EAST nemzetközi projekt keretében dolgozták fel.

már feldolgozott 1984 c. regényt a projekt keretében saját módszerekkel újra elemeztük.)

3. Újságnyelv

Magyar napilapok, ill. heti újságok közül a Magyar Hírlap 1999. január 4-i, a Népszabadság 1999. április 3-i, a Népszava 1999. november 25-i, és a HVG 1999. szeptember 4-i teljes száma került a korpuszba. A napi- és hetilapok számai a Népszava kivételével az újságok archív CD-ROM-jairól származnak, a Népszava cikkei pedig az újság internetes honlapjáról valók[6].

4. Számítástechnikai szövegek

Számítástechnikai szövegek képviseltetése céljából az IDG kiadóvállalat *ComputerWorld Számítástechnika* c. újságjának összegyűjtött számai, valamint Kis Balázs: Windows *2000 – haladó könyv haladó szoftverhez* c. könyvének 4., 5., 6. fejezete szerepelnek a korpuszban. A korpusznak ez a része az Interneten gyakran előforduló számítástechnikai, technológiai jellegű szövegeket reprezentálja.

5. Törvényszövegek

Jogi témájú szövegek közül az 1997. évi 144. törvény *A gazdasági társaságokról*, illetve az 1999. évi 76. törvény *A szerzői és szomszédos jogokról* teljes szövege került be a korpusz anyagába. A szöveg a *CD Jogtár: Hatályos magyar jogszabályok* CD-ROM-ról származik.

6. Rövid pénzügyi és gazdasági hírek

A különböző üzleti, tőzsdei, piaci és egyéb gazdasági témájú szövegek reprezentálása céljából az MTI-ECO gazdasági hírszolgálat *Business Plus* nevű adatbázisából származó hírek is bekerültek a korpuszba. Az anyag a 2001. márciusától 2002. januárjáig terjedő időszak híreit öleli fel. A hírek összeválogatásánál fontos szempont volt, hogy egyetlen bekezdésből álljanak. A korpusz összefoglaló adatai az 1. táblázatban láthatók.

Törekedve arra, hogy a korpusz nemzetközileg elfogadott szabványoknak megfeleljen, a szövegek reprezentációja XML-formátumú[7], a leíró séma pedig a TEI DTD-nek[8] felel meg.

3. Morfológiai elemzés a Szeged Korpuszon

A korpusz nyelvi feldolgozásához, ill. a morfológiai elemzés megvalósításához a szövegeknek bizonyos előkészítő munkálatokon kellett átesniük. A feldolgozás egyik előfeltétele a szövegek mondatokra, majd szavakra bontása, vagyis a szöveg szavainak, nyelvi jeleinek szegmentálása, a tokenek kialakítása volt (pl. az írásjelek leválasztása az előtte álló szóról, több szóból álló tulajdonnevek megjelölése stb.). Ezután a kapott tokenekből alakult ki a szótár, amelynek címszavai tehát nem a szótövek, hanem az egyes előforduló szóalakok (jeles, ragos formák). A szótár elkészülte után a korpusz tokenjei mellé morfo-szintaktikai kódok kerültek.

---

[6] A Népszava honlapjának címe: http://www.nepszava.hu/.

[7] Az XML-honlap címe: http://www.xml.org

[8] A TEI DTD leírása az alábbi honlapon található meg: http://www.tei-c.org

Az így előállt szótárban a szófaji előelemzést a MorphoLogic Kft. által kifejlesztett HuMor magyar szófaji elemző programmal [6] végezte a csoport. A korpusz hivatalos szófaji kódolására azonban nem a HuMor által megadott kódokat, hanem az MSD-(Morpho-syntactic Description)-kódrendszer [3] magyar változatát használták, de ezzel párhuzamosan tárolásra kerültek a generált HuMor-kódok is. A HuMor-MSD közti megfeleltetést egy konvertáló programmal biztosították. Ennek a programnak a felhasználásával lehetett előállítani a még nem egyértelműsített, de a szófaji elemzés eredményét, vagyis minden lehetséges szófaji kódot tartalmazó korpuszfájlokat. Ha egy szóalak többjelentésű volt, és a jelentéskülönbségek morfológiailag megfogalmazhatók voltak, akkor egy-egy szócikkhez ennek megfelelően több MSD-kódot is tartozhatott. Az előkészítő munkálatok nyelvészeti vezetője Dr. Bibok Károly, a Szegedi Tudományegyetem Orosz Filológiai Tanszékének docense volt.

A fenti előkészületek után az egyetem 20 nyelvészhallgatója kézi ellenőrzés és javítás formájában végezte az egyértelműsítési munkát az előelemzett korpuszon. Az egyértelműsítés során a több kóddal rendelkező szavak esetében eldöntötték, hogy a lehetséges kódok közül melyik érvényes az adott szövegkörnyezetben, és ezt helyes kódként jelölték meg. A szófaji elemzésben és egyértelműsítésben felmerülő problémák megoldásakor a Magyar Értelmező Kéziszótár (1972)[9] volt az irányadó.

Az egyértelműsítési munka eredményeként elkészült az 1 millió szavas (első öt szövegcsoport) kézzel annotált magyar nyelvű tanuló adatbázis, amely egy automatikus szófaji egyértelműsítő módszer kialakításának alapja, és amely a további természetes nyelvi feldolgozáshoz kiindulásként szolgál. [2], [4], [5]

4. Szintaktikai elemzés a Szeged Korpuszon

A szintaktikai elemzést több lépcsőben végzi a csoport. Első lépéseként a főnévi csoportok bejelölését vállalta. A választás nyelvészetileg is megindokolható, hiszen a mondatot jellemzően egy ige (vagy más predikatív csoport), és annak (kötelező és szabad) bővítményei alkotják, a bővítmények pedig jellemzően főnévi csoportok. A magyar nyelvben ezek a bővítmények nemcsak nyelvtani szempontból játszanak központi szerepet, de a legtöbb esetben tartalmilag is ezek hordozzák a legfontosabb információt. Ezt a kezdeti elemzési lépést a nemzetközi szakirodalom lapos szintaktikai elemzésként (shallow parsing) azonosítja.

Definíció szerint a főnévi csoport (<NP> és </NP> címkék között) egy adott mondaton belül előforduló, egy vagy több szót, ill. írásjelet magába foglaló szerkezet, amelyre a következő feltételek teljesülnek:

- A csoport feje főnév (vagyis a csoportban szereplő utolsó szó főnévi szófajú);
- A csoport folyamatos (vagyis a szerkezetben szereplő szó/szavak, ill. írásjel/ek egymást követik);
- A csoport mérete maximális (vagyis nem lehet további elemmel bővíteni).

A főnévi csoportok lehetnek egymásba ágyazottak, de nem keresztezhetik egymást.

---

9 Juhász J., Szőke I., Nagy G. O., Kovalovszky M. (ed.): Magyar Értelmező Kéziszótár Akadémiai Kiadó, Budapest, (1972)

A morfológiai előfeldolgozáshoz hasonlóan a főnévi csoportok bejelöléséhez is egy automatikus előelemző programot használt a csoport. Az előelemzés előre meghatározott reguláris szabályok alapján a CLaRK rendszer10 segítségével történt. Ezt kézi szakértői ellenőrzés és javítás követte. Ott, ahol az előre meghatározott reguláris szabályok nem fedték le a főnévi szerkezeteket, a kézi annotálók speciális, a szerkezetre vonatkozó információval bővítették az annotálást. A munkálatok eredményeként előállt a korpusz elemzésének második szintje, amely a későbbiekben egyrészt gépi tanuló algoritmusok tanulási adatbázisaként szolgált, másrészt a természetes nyelvi mondatok szemantikai elemzésének kiindulópontja. [2], [4]

A következő lépésben a tagmondathatárokat keresték meg és jelölték be az annotálók. A (teljes) mondatban található tagmondatokat egy nyitó <CP> és egy záró </CP> címkével jelölték. A tagmondatok bejelölésénél egyrészt meg kellett határozni magukat a tagmondatokat, másrészt el kellett helyezni azokat az egész mondatszerkezetben. A tagmondatok alárendeltek vagy mellérendeltek lehetnek. Az alárendelt tagmondatok teljes egészében benne találhatóak egy másik (tag)mondatban, a mellérendelt tagmondatok együtt (egyenrangúan) alkotnak egy bővebb tagmondatot. Csak azok a kötőelemek tartoznak a tagmondatba, melyek abban mondatrészi szerepet töltenek be. Így pl. a hogy kötőszó nem része az alárendelt tagmondatnak, a vonatkozó névmások viszont igen.

5. Folyó és jövőbeli mukák

A természetes nyelvi feldolgozó csoport jelenleg folyó munkálatai egy bővebb szintaxis (*treebank*) kialakítására irányulnak. Ez egyrészt az igéknek, az igeneveknek és a főnévi csoportként nem azonosított mondatszintű bővítményeknek (határozószói csoportok, névutós csoportok, elváló igekötők) az annotálását, másrészt a főnévi csoportok belső szerkezetének finomítását jelenti.

A főnévi csoportként nem azonosított mondatszintű bővítmények bejelölése a következő irányelvek alapján történik. Névutós csoportot (PP: postpositional phrase) akkor kapunk, ha főnévi csoporthoz névutót kapcsolunk. A névutó általában az NP után áll, de ritkán meg is előzheti (Pistával szemben, szemben Pistával). Segítség: a névutók MSD-kódja: St.

A határozószói csoportok (ADVP: adverbial phrase) közé tartoznak a határozószók (*most, itt, tavaly, mindig* stb.; MSD-kódjuk első karaktere jellemzően: R), a „határozóragos" melléknevek (*betegen, angolul, jogilag* stb.; MSD-kódjuk jellemzően: Afp-sw) és számnevek (*öten, sokan* stb.; MSD-kódjuk: M?-sw?), és a „névutós" személyes névmások (*mellettem, miattad, utánunk* stb.; MSD-kódjuk: Rl--??). Nem tartoznak ide a határozói igenevek (-va/-ve; -ván/-vén; MSD-kódjuk: Rv). Jellemző mondattani szerepük: igei jellegű rész határozói bővítménye (*tegnap* jött), vagy fok- mértékhatározó (*nagyon* finom).

---

[10] A CLaRK rendszert Kiril Simov, a Bolgár Tudományos Akadémia dolgozója fejlesztette ki a BulTreeBank projekt keretében (http://www.bultreebank.org).

Az elváló igekötők (pl: meg, el, be, stb.) bejelölése a <PREVERB> és </PREVERB> címkék segítségével történik. Meghatározásukban az igekötők MSD-kódja (Rp) lehet iránymutató.

A treebank kialakításának másik fontos eleme a főnévi csoportok belső szerkezetének finomítása, azaz a melléknévi jelzők bejelölése. A melléknevek (ADJP: adjective phrase) minőségjelzői szerepben találhatók, rendszerint egy NP fejének (főnevének) jelzőjeként, a *milyen?* vagy a *melyik?* kérdésre válaszolva. A predikatív melléknevek NP-ként már be vannak jelölve, így ezeket nem jelöljük melléknévi szerkezetként, ill. ha a melléknév egy NP feje, akkor sem kap <ADJP>-címkét (pl. *az első* jöjjön be). Az ADJP-knek lehet bővítményük (leginkább ADVP), ill. mellérendelés is előfordulhat a melléknevek között, ha a két melléknév egyenrangú. Ez azt jelenti, hogy akár fel is cserélhető a sorrendjük, és tartalmilag így is ugyanaz a jelentésük marad (pl: szép és okos gyerek, okos és szép gyerek). Két egymás mellé rendelt ADJP újabb ADJP-szerkezetet alkot.

A fenti szintaktikai annotálás eredményét az alábbi példamondat szemlélteti:

```
<CP>
        <ADVP>
                Ekkor
        </ADVP>
        <NP>
                egy
                <ADJP>
                        <ADVP>
                                olyan
                        </ADVP>
                        hangos
                </ADJP>
                jármű
        </NP>
        robogott
        <PREVERB>
                be
        </PREVERB>
        ,
        hogy
        <CP>
                megfájdult
                <NP>
                        tőle
                </NP>
                <NP>
                        a
                        fülünk
                </NP>
        </CP>
</CP>
```

Az igei csoportok bejelölése ugyanazt a stratégiát követi, mint a morfológiai elemzés és a főnévi csoportok bejelölése. Az annotátorok feladata ezúttal is egy automatikus előelemzés eredményének ellenőrzése, ill. javítása lesz. Az annotálás eredményeként a szövegben előforduló igék és igenevek kerülnek bejelölésre, majd a szövegkörnyezetnek megfelelő hozzájuk tartozó vonzatkeretek egy táblázatból kerülnek kiválasztásra. A táblázatot egy domain-ontológia egyszerűsített változataként kell elképzelni, amely az egyes igék kötelező vonzatainak szemantikai attribútumait tartalmazza.

A csoport távolabbi terveiben szerepel egy bővebb ontológiai adatbázis (hierarchikus fogalomháló) kialakítása is, amely elsősorban a szemantikai információk annotálásában, a szemantikai szerepek azonosításában, és az ezek alapján történő információkinyerésben játszik fontos szerepet. A szemantikai annotálás során a különböző tartalmi elemek azonosítása és kódolása történik, ún. szemantikai keretek segítségével. [4] Ezt az említett NKFP projekt keretében a konzorcium tagjai meg is valósították a 200 ezer szavas rövid híreket tartalmazó szövegállományon. A továbbiakban a fennmaradó 1 millió szón végzi el az egyetem természetes nyelvi csoportja ugyanezt a munkát. A szemantikai keretek olyan kötelező, ill. szabad alegységek sorozatából állnak, amelyek ráilleszthetők a morfológiailag és szintaktikailag elemzett szöveg elemeire. Az alegységek ezáltal olyan információkkal telnek meg, amely alapján megbízhatóbban lehet válaszolni egy információszerzés céljából feltett kérdésre. Ha például valaki egy konkrét cég felvásárlása után érdeklődik, akkor a CÉG ADÁSVÉTELE szemantikai keretet felépítő VEVŐ, ELADÓ, CÉG, ÖSSZEG alegységekben szereplő információ alapján választ kaphat kérdésére.

Irodalomjegyzék

1. Alexin Z., Csirik J., Gyimóthy T., Bibok K., Hatvani Cs., Prószéky G., Tihanyi L.: Manually Annotated Hungarian Corpus in Proc. of the Research Note Sessions of the 10th Conference of the European Chapter of the Association for Computational Linguistics EACL'03, Budapest, Hungary 15-17 April, (2003) pp. 53-56
2. Alexin Z., Váradi T., Pravecz Cs., Prószéky G., Csirik J., Gyimóthy T.: FGT – A Framework for Generating Rule-based Taggers. ILP-99 Late-Breaking papers, Bled, Slovenia 24-27 June, (1999) pp. 1-7
3. Erjavec, T., Monachini, M., (ed.): Specification and Notation for Lexicon Encoding. Copernicus Project 106 „MULTEX-EAST", Work Package 1 – Task 1.1, Deliverable D1.1F, (1997)
4. Hócza A., Alexin Z., Csendes D., Csirik J., Gyimóthy T.: Application of ILP methods in different natural language processing phases for information extraction from Hungarian texts in Proc. of the Kalmár Workshop on Logic and Computer Science, Szeged, Hungary, 1-2 October (2003), pp. 107-116
5. Horváth T., Alexin Z., Gyimóthy T., Wrobel S.: Application of Different Learning Methods to Hungarian Part-of-Speech Tagging in Proc. of the 9th International Workshop on Inductive Logic Programming (ILP99) Bled, Slovenia, int he LNAI series vol 1634, pp. 128-139, Springer Verlag (1999)
6. Prószéky G., Kis B.: A Unification-based Approach to Morpho-syntactic Parsing of Agglutinative and Other (Highly) Inflectional Languages in Proc. of the 37th Annual Meeting of the Association for Computational Linguistics, College Park, Maryland, USA (1999), pp. 261-268

Manually Annotated Hungarian Natural Language Corpus: the Szeged Korpusz

Dóra Csendes[1], Csaba Hatvani[1],
Zoltán Alexin[1], János Csirik[1], Tibor Gyimóthy[1], Gábor Prószéky[2], Tamás Váradi[3]

[1] University of Szeged, Department of Informatics
H-6720 Szeged, Árpád tér 2., Hungary
{dcsendes, hacso, alexin, csirik, gyimi}@inf.u-szeged.hu
http://www.inf.u-szeged.hu
[2] MorphoLogic Ltd. Budapest
H-1118 Budapest, Késmárki u. 8., Hungary
proszeky@morphologic.hu
http://www.morphologic.hu
[3] Research Institute for Linguistics at the Hungarian Academy of Sciences
H-1068 Budapest, Benczúr u. 33., Hungary
varadi@nytud.hu
http://www.nytud.hu

Keywords: natural language corpus, morpho-syntactic annotation, shallow parsing, treebank building

The present state of the Szeged Korpusz is the result of three national projects and the cooperation of the following three consortium partners: the University of Szeged, Department of Informatics, MorphoLogic Ltd. Budapest, and the Research Institute for Linguistics at the Hungarian Academy of Sciences.

The corpus currently comprises 1.2 million words plus 290 thousand punctuation marks. Texts have gone through different phases of natural language processing and analysis. First, they were segmented into manageable units, i.e. words, punctuation marks and special tokens.

In the second step, corpus words were morpho-syntactically analysed with the help of an automatic analyser (the HuMor Hungarian morphological tagger developed by MorphoLogic Ltd.) and then manually POS tagged by linguistic experts.

In the following phase of processing, texts of the Szeged Korpusz have been shallow parsed. During this phase annotators marked noun phrase structures and the clause structure of corpus sentences.

Current works aim at a more detailed syntactic analysis of the texts including the annotation of adverbial, postpositional, adjectival structures and the identification of verbs and their argument structures. With this, we intend to lay the foundation of a Hungarian treebank that – as a continuation of the work – is planned to be enriched with semantic information as well.

The Szeged Korpusz is publicly available after on-line registration (http://www.inf.u-szeged.hu/lll) and can be used free of charge for educational and research purposes.

A MetaMorpho projekt története

Tihanyi László

MorphoLogic Kft. 1118 Budapest Késmárki utca 8.
tihanyi@morphologic.hu

Absztrakt. A MetaMorpho fordítóprogram-család a fordítási feladatok teljes palettájára eszközöket kíván biztosítani. A nyelvet nem tudók számára megértéstámogató eszköz, az átlagos nyelvtudásúak számára fordítóprogram, és a profi fordítóknak intelligens fordítómemória készül.

1. Előzmények

A MorphoLogic 1991-es megalapítása óta foglalkozik egy gépi fordítási projekt beindításának gondolatával. A konkrét munka elkezdéséhez azonban meg kellett teremteni a szükséges anyagi és technikai hátteret. Sokáig úgy terveztük, hogy egy már meglévő fordítóprogramnak készítjük el a magyar modulját, és ezért korábban kapcsolatba léptünk több gépi fordító rendszer készítőjével (Logos [1], Systran [2] és LogoVista [3]). Végül – és ma már tudjuk: szerencsésen – a saját rendszer fejlesztése mellett döntöttünk.

2. 2000

A tulajdonképpeni munka 2000. júniusában kezdődött el. Először meghatároztuk az általunk kifejlesztendő rendszer működési elvét. [4, 5] Eszerint a MetaMorpho olyan szabály-alapú rendszer, ahol az elemzési szabályokhoz közvetlenül hozzárendelt fordítások tartoznak. Az így felírt mintapárok tetszőlegesen specifikált elemekből állhatnak, amelyeket ettől függően nyelvtani szabálynak, lexikális elemnek, vonzatkeretnek, vagy egyéb mintának nevezhetünk. Ha a teljes mondat minden szavát lexikálisan megszorítva tároljuk el, akkor a mintaalapú rendszerek működését valósítjuk meg, azaz fordítómemória-szerű működést nyerünk. A MetaMorpho ezek szerint egyszerre RBMT (Rule-Based Machine Translation) és EBMT (Example-Based Machine Translation), azaz egyesíti az MT (Machine Translation) és a TM (Translation Memory) rendszerek tulajdonságait.

Nem köteleztük el magunkat semmilyen ismert formális nyelvészeti elmélet mellett, környezetfüggetlen szabályokat használunk, praktikus kiegészítésekkel. Az elemzés balról jobbra halad, és bottom-up irányban építkezik.

A MetaMorpho kétnyelvű eszköz. Az elemzés eredménye a felcímkézett fa, a generálás a fa top-down bejárása az elemző szabályok generáló párjainak

kiértékelésével. A nyelvtan egyirányú, és bár a minták szimmetrikusak, ez a projekt kizárólag az angolról magyarra fordításra korlátozódik

Az egyszerű architektúra célja a hatékony bővíthetőség. A mintákra épülő rendszer képes arra, hogy tudását futási időben lehessen bővíteni, és a így szerzett ismeretet a sajátjával azonos módon kezelje.

Szeptemberben kialakítottuk az MMO-szabályok szintaxisát. Meghatároztuk az elemzési szimbólumokat, ezek tulajdonságait és értékkészletét. Az ötlet, hogy a szabályokat Clips-ben, egy Lisp-alapú szakértői rendszerben működtessük, Endrédy Istvántól származott. Kezdetben Tihanyi László végezte a rendszer tervezését és kialakítását, Vlaskovits Dóra a nyelvtani szabályok írását, Endrédy István pedig a szakértői rendszer programozását. A Clips azért bizonyult jó választásnak, mert saját elemzőrendszer írása nélkül meg lehetett bizonyosodni arról, hogy a tervezett rendszer működőképes. Ennek az volt az ára, hogy a szabályokat a Lisp által meghatározott formátumúra kellett konvertálni.

A találatok számának csökkentése érdekében még ez évben bevezetésre került egy mechanizmus, amely biztosította, hogy a lexikálisan kitöltöttebb, azaz specifikusabb minták felül tudják bírálni a velük azonos hosszon illeszkedő általánosabbakat. Ezzel el lehetett érni, hogy a rendszer gyakorlatilag mindig csak egy elemzési eredményt adjon.

Az első morfológiai generátornyelvtant – épp a MetaMorphóhoz – októberben készítette el Tihanyi László és Endrédy István. Az év végére kialakult a MetaMorpho laboratóriumi prototípusa.

3. 2001

Az első problémák akkor jelentkeztek, amikor a szabályaink bonyolódtak és a működés nyomkövetése a szakértői rendszer formalizmusában egyre nehézkesebbé vált. Emiatt a lehetővé tettük, hogy a szabályírás a már induláskor felvázolt MMO szintaxisban történjen. Ekkor, márciusban jött létre a szabályok szintaxisának specifikációja. [6]

A Clips-nyelvtanban írt mintáinkat Vlaskovits Dóra áttette saját formalizmusunkba. Tihanyi László konvertereket készített, amelyek a MMO formátumú mintákat először XML-re majd azt Clips-nyelvűre fordították. [7]

Májusban meghatároztuk a rendszer moduljait és a fordítási lépéseket: morfológiai elemzés, morfológiai elemek szintaktikai szimbólumokká konvertálása, szintaktikai elemzés és generálás, szintaktikai elemek morfológiaivá konvertálása, morfológiai generálás. A modulokat szakértői rendszerben is kialakítottuk, és a környezetbe bekötöttük a *Humor* [8] morfológiai elemzőt és generátort. A morfológiai és szintaktikai szimbólumok konverziója ekkor még Lisp nyelven történt.

Készítettünk egy – mintegy húszezer elemű – angol-magyar szótárat, amelynek fontos tulajdonsága volt, hogy minden szónak csak egy jelentése lehetett.

A rendszer 2001. májusára valódi fejlesztő környezetté nőtte ki magát, amelyből a szabályok, az akkor még különálló szótárak és a morfológia is bővíthetővé vált.

A saját formátum bevezetése még nem jelentett megoldást a nyomkövetésre. Kezdte éreztetni a hatását az eredetileg nyilván más célra készült szakértői rendszer

korlátoltsága. A szabályok számának növekedésével a rendszer lelassult. Kis Balázs megkezdte a kialakított adatszerkezetnek és működési elvnek megfelelő szintaktikai elemző, a *HumorESK* [9] fejlesztését.

Az év második felét a nyelvi adatok gyűjtésére fordítottuk. Ugray Gábor először szótárfejlesztési feladatokkal lett megbízva: több szótár összevetésével kiválasztotta az angol legvalószínűbb magyar jelentését. Közben MMO-formátumban elkészült mintegy tizenötezer angol ige vonzatkeret-leírása. Új kódokat kaptak továbbá a magyar névszók is, az eddig még nem kezelt „hol/honnan/hová" kérdésekre adott morfológiai viselkedésük alapján.

A nagytömegű Clips-minta előállítása az XSLT processzorral már kezdett lassúnak bizonyulni, ezért ezt a konverziót DOM-alapú C++ programmal kellett kiváltani. Még az év vége előtt összekötöttük a programot a *MoBiMouse* felhasználói felületével.

4. 2002

Először elvégeztük a morfológiai és szintaktikai modulok összekötését, melyből egy véges állapotú automatával működtetett konverter született. Időközben Endrédy István létrehozta a C nyelvű API-t, így ezzel a modulok önálló programmá váltak és megvalósulhatott a belső adatforgalom. A elemzőmodulok között ezután Tihanyi László XML interfészt definiált: a *format.dtd*-ket.

Az elhúzódó HumorESK-fejlesztés miatt további elemző-fejlesztések is elindultak a cégen belül. A versenyt Ugray Gábor *moose* elemzője „nyerte meg". Jelenleg is ez alkotja a MetaMorpho motorját: a *moose* egyben a köré épült nyelvtanfejlesztő környezet neve is. Nyílt elemzői felületünk lehetővé teszi, hogy más elemzők (pl. a HumorESK) a MetaMorpho rendszerbe illeszkedjenek. Az új elemző új lehetőségeket teremtett a nyelvtanban is, amely ennek megfelelően átdolgozásra került. Ilyenek a voltak: részfa-mozgatás, vagy a nem-terminális elemek beszúrása. Az új rendszer közvetlenül olvassa be az MMO-szabályokat: ezzel véget ért a Clips-korszak.

Ezután bevezettük a források kétszintűségét: létrejött az MMD-formátum. Ez egy MMO-val szintaktikailag azonos forma, mely az olvashatóság érdekében nem tartalmazza azokat a tulajdonságokat, amelyek triviálisan gépi úton is hozzájuk rendelhető (pl. az öröklődő tulajdonságok).

Endrédy István elkészítette a MetaMorpho első telepíthető változatát, a MoBiMouse-felületű *MoBiCAT* programot. Közben a projekt nyelvi- és program-forrásainak archiválására, illetve az egyidejű projektmunkára bevezettük a CVS-rendszert. Bevezettük a projekt belső ellenőrzését is: a szabályok működését a hozzájuk tartozó mintamondatokkal rendszeresen ellenőrizni kezdtük. Augusztusban elindítottuk a szabályok egymás közti és a morfológiai rendszerrel való konzisztenciájának ellenőrzését.

Az ismeretlen angol szavak magyar toldalékolására Novák Attila egy *guesser*-programot fejlesztett ki. [10]

Az igei vonzatkeretek fordítása májustól októberig, azaz fél évig tartott; tesztelése még ma is folyik. A lexikonok is fejlődtek: kifejezés-, névszóielőtag- és más tematikus tulajdonnév-lexikonok készültek.

5. 2003

A felmerült tennivalók kezelésére januárban bevezettünk egy intranetes célprogramot az mmotodo *bugz* rendszert. Ezzel egy időben Gröbler Tamás került a fordítómemória-projekt élére, mely ettől új lendületet kapott: először elkészítette a C++ alapú objektumorientált interfészt, amely a régi C-alapú XML API-t váltotta ki. Emellett több új modul is született, mint például a szövegfájlokon futtatható *documentTranslator*, vagy a szabályok bővítésére készített TM-interfész, Hodász Gábor pedig megoldotta a fordítómemória-bejegyzések tárolását mySQL adatbázisban.

Időközben Ugray Gábor elkészített egy új MMD–MMO konvertert, mely a szabálybővítéseket volt hivatott áttekintehetőbben kezelni. A nyelvtani leírás is folyamatosan bővült: Újvárosi Gábor végzett az időhatározós szerkezetek kódolásával [11]

Júniusban Kundráth Péter először a HTML oldalak fordítását oldotta meg, majd elkészítette a MetaMorpho szerverváltozatát, HTML- és WAP-kliensekkel. A nyáron tovább bővült a csoport Vlaskovits Dórával és Merényi Csabával, akik az alapnyelvtan fejlesztéséért felelősek.

Szeptemberben Újvárosi Gábor átfogó felmérést készített a nyelvtan állapotáról egy hagyományos angol-magyar nyelvtankönyv, a *Huron's Checkbook* [12] alapján. Ezt követően megindult a mostantól már folyamatosan folyó tesztelés, Vancsa László vezetésével. Ő a rendszeresített teszteljárások mellett bevezette az anyag Bleu-teszttel történő vizsgálatát. [13,14]

A Humor-rendszerben működő angol morfológia könnyebb belső kezelésére létrehoztunk egy listaformátumot, amellyel a karbantartás hatékonyabbá vált. Októberben Kundráth Péter és Endrédy István közreműködésével létrejött a második telepíthető MetaMorpho-alkalmazás: az *MmoServer*.

6. A jövő

Célunk, hogy a MetaMorpho olyan fordítóprogram-család legyen, amely a fordítási feladatok teljes palettáján használható eszközöket biztosít: a nyelvet nem tudók számára megértéstámogató eszköz, az átlagos nyelvtudásúak számára fordítóprogram, és a profi fordítóknak fordítómemória-változat készül. A program első, széles körben használható változata az uniós csatlakozásra megjelenő *MoBiCAT* mondatszintű megértéstámogató–fordító eszköz lesz.

A jövő májusig terjedő időszak főbb fejlesztései:

- Többszálú szintaktikai elemző
  A *moose* szintaktikai elemzőt Ugray Gábor alakítja át az internetes felhasználásnak is eleget tevő, többszálú üzemmódra.
- Kliens–szerver fordítórendszer
  A *Microsoft Word*-ön belüli fordítást támogató, szervermegoldásra épülő fordítóprogram első változatán Kundráth Péter dolgozik.

- Automatikus mintageneráló
  Vajda Kristóf szeptemberben megkezdte a szabálybővítő program és felhasználói felületet fejlesztését, melytől a mintabővítés hatékonyságának drasztikus javulását várjuk.
- Szabálykonverter
  Hegedűs Balázs végzi a szabályforrásaink bővítését végző konverter [15] újraírását.
- Szövegszinkronizáló
  Nyár óta folyik a szövegszinkronizáló program fejlesztése Pohl Gábor vezetésével. Októberre elkészült az eszköz motorja, és megindult a felhasználói felület fejlesztése. A modul segítségével fordításokból, párhuzamos szövegkorpuszokból hatékonyan tudunk majd mintatárat készíteni.
- Jelentés-egyértelműsítő
  Miháltz Márton kutatásai eredményeként a 2003-as év végére elkészül a szavak helyes jelentésének kiválasztását végző jelentés-egyértelműsítő eszköz.

7. Összegzés

A MorphoLogic cég 1991-es alapításától kezdve készített minden fontosabb nyelvtechnológiai modul felhasználásra került a MetaMorpho-projektben. A 2000. júniusa és 2003. decembere között zajló fejlesztési időszakban mintegy 30 ember működött közvetlenül közre, ami kb. 220 ember-hónapot jelent. A munka fele-fele arányban oszlik meg a programozás és a nyelvtanfejlesztés között: a program jelenleg mintegy 70 modulból (MSVC-projektből) áll, adatbázisa 110 ezer szabályt tartalmaz.

A projekten ma 13 fő dolgozik. Ezen kívül sok segítséget kap a MetaMorpho-csapat a cég többi munkatársától, illetve a kutatás-fejlesztéshez kapcsolódó egyetemistáktól és doktorandusz-hallgatóktól. A teljes MetaMorpho-projektet kezdektől fogva a MorphoLogic finanszírozza saját erőből.

Referenciák

1. Hawes, R.E.: Logos: The Intelligent Translation System. In: Lawson, V. (ed.) *Tools for the Trade. Proceedings of the Confetrence 'Translating and the Computer 5'.* London: Aslib, 131-139 (1985)
2. Toma, P.: Systran as a Multilingual Machine Translation System. In: *Overcoming the language barrier. Third European Congress on Information Systems and Networks,* Luxembourg,. München: Verlag Dokumentation, 569-581 (1977)
3. Akers, Glen: Logo Vista Conquers Japan *Language Industry Monitor.* 1994 Mar-Apr (1994)
4. Tihanyi László: A fordítóprogram működése *MorphoLogic belső dokumentáció* (2000. június)
5. Tihanyi László: A MetaMorpho formátum. *MorphoLogic belső dokumentáció* (2000. szeptember)

6. Tihanyi László: A MetaMorpho felépítése. *MorphoLogic belső dokumentáció* (2001. május)
7. Tihanyi László: A mmorpho2.dtd. *MorphoLogic belső dokumentáció* (2001. március)
8. Prószéky Gábor & Tihanyi László: A Fast Morphological Analyzer for Lemmatizing Corpora of Agglutinative Languages. *Papers in Computational Lexicography.* Linguistics Institute of H.A.S, 265–278 (1992)
9. Prószéky Gábor: Syntax As Meta-Morphology. *Proceedings of COLING-96,* Vol.2, 1123–1126. Copenhagen, Denmark (1996)
10. Novák Attila, Nagy Viktor, Oravecz Csaba: Magyar ismeretlenszó-elemző program fejlesztése. *I. Magyar Számítógépes Nyelvészeti Konferencia, Szeged* (2003)
11. Ugray, Gábor & Gábor Ujvárosi: English Adverbial NPs of Time in Machine Translation *Proceedings of RANLP*, Tzigov Chark, Bulgaria (2003)
12. Salamon Gábor, Zalótay Melinda (szerk..): *Huron's Checkbook 8000.* Műszaki Könyvkiadó (2001)
13. Papineni K., Roukos S., Ward T. & Zhu W.-J. BLEU: a Method for Automatic Evaluation of Machine Translation. *Research Report,* Computer Science IBM Research Division, T.J.Watson Research Center (2001)
14. Vancsa László: A „BLEU" automatikus kiértékelési eljárás alkalmazása angol-magyar fordítóprogram gyakori, folyamatos minősítésére. *I. Magyar Számítógépes Nyelvészeti Konferencia, Szeged* (2003)
15. Hegedűs Balázs: Szabálykonverzió a MetaMorpho rendszerben. *Szakdolgozat.* BME Villamosmérnöki Kar (2003)
16. Prószéky, Gábor & László Tihanyi: MetaMorpho: A Pattern-based Machine Translation Project. In: *Proceedings of the 24th 'Translating and the Computer' Conference.* London, United Kingdom, 19–24 (2002)

Függelék: A fejlesztésben közreműködő munkatársaink

Aggod Zsuzsa, Csordás Attila, Dominus Ákos, Endrédy István, Földes András, Grőbler Tamás, Hegedűs Balázs, Hodász Gábor, Keresztes Máté, Kis Balázs, Kiss Gabi, Kiss Márton, Kozma Andrea, Kundráth Péter, Légrádi Ágnes, Magyar Dóra, Merényi Csaba, Miháltz Márton, Novák Attila, Pál Miklós, Pohl Gábor, Prószéky Gábor, Tihanyi László, Tőkés Tamás, Ugray Gábor, Újvárosi Gábor, Vancsa László, Vajda Kristóf, Vlaskovits Dóra

The Story of MorphoLogic's MetaMorpho Project

László Tihanyi

MorphoLogic
1118 Budapest Késmárki u. 8.
tihanyi@morphologic.hu

Keyword

Machine translation (MT), translation memory (TM), computer-aided translation (CAT)

Abstract

MetaMorpho is a machine-translation program family supporting different areas of translation: comprehension assistance for beginners, MT for standard users and TM for professional translators.

Development of MetaMorpho began in 2000, using all the language technology results of MorphoLogic achieved in the previous 10 years. Since then there have been 30 programmers and linguists working on the project. It means about 200 more man-month for this period. At the beginning there were three developers – and now we have thirteen full-time people and several PhD students.

MetaMorpho is a combination of rule-based and example-based systems. We use the term pattern for both linguistic rules and lexical entries depending on their specification level. MetaMorpho thus unifies the advantages of the RBMT and EBMT systems.

The first aim of this project is a running English-Hungarian translation module. Since the basic idea behind the software tool is language independent, new language-pairs can be developed relatively easily.

Presently, we are working on a tool improving efficiency of grammar development.

The first publicly available results of this 4-year project are going to be published next year. The very first product called MoBiCAT is a higher-level version of the intelligent comprehension assistant, the popular MoBiMouse.

Szövegszinkronizációs módszerek, hibrid bekezdés- és mondatszinkronizációs megoldás

Pohl Gábor

Pázmány Péter Katolikus Egyetem Információs Technológiai Kar
pohl@morphologic.hu

Kulcsszavak: szövegszinkronizáció (text alignment), mondatszinkronizáció (sentence alignment), bekezdésszinkronizáció (paragraph alignment), statiszikai módszerekkel szűrt horgonyok, hibrid szinkronizációs megoldás.

Kivonat: A cikkben bemutatjuk a szövegszinkronizáció általánosított definícióját; a szövegegységek hosszán alapuló szinkronizációt; a horgonyok használatának és statisztikai szűrésének lehetőségét; majd olyan, a két stratégiát ötvöző hibrid szinkoronizációs megoldást ismertetünk, amely beszúrásokat és elhagyásokat tartalmazó szövegpárok szinkronizálására alkalmasabb az eddigi módszereknél.

1 Szövegszinkronizáció

Szövegszinkronizáción (text alignment) két- vagy többnyelvű szövegekben az egymás fordításának tekinthető szövegegységek meghatározását értjük. Fordítómemória nélkül készített fordítások fordítómemóriákba töltéséhez, fordítások terminológiai konzisztenciájának gépi ellenőrzéséhez, nyelvészeti kutatások alapjául szolgáló párhuzamos korpuszok építéséhez legalább mondatszintű szinkronizációra van szükség. Az egymás fordításának tekinthető mondatok gépi meghatározása azonban nehéz feladat, mivel a fordítók a mondatok határait a fordítás során megváltoztathatják, (véletlenül) elhagyhatnak, illetve beszúrhatnak mondatokat, felcserélhetik a mondatok sorrendjét.

1.1 Általánosított definíció

A szövegszinkronizáció definícióját korábbi szerzők saját munkáikban különbözőképp határozták meg, hiszen az általuk megoldani kívánt szinkronizációs feladatok, is különbözők voltak. A következőkben az általunk használt általánosított definíciót [1] mutatjuk be.

Szövegszinkronizáción párhuzamos szövegek szinkronizációs egységeinek olyan egy-egyértelmű egymáshoz rendelését értjük (szinkronizációsegység-párokat határozunk meg), amely során egy adott forrásszövegbeli szinkronizációs egységhez egy

fordításbeli szinkronizációs egységét csak akkor rendelünk hozzá, ha az a teljes szövegeket tekintve a forrásszövegbeli szinkronizációs egység fordításának tekinthető.

Szinkronizációs egységen általánosságban a szöveg pontosan meghatározott típusú egységeinek – nem feltétlenül egymáshoz tapadó egységekből képzett – halmazát érjük, amely nem tartalmaz más szinkronizációs egységben szerepelő szövegbeli egységet. Ez a definíció tehát megengedi, hogy például két egymástól távol eső mondat (konkrét szövegbeli előfordulás) alkosson egy szinkronizációs egységet, de a szinkronizációs egységek között nem lehet átfedés, azaz az adott mondatoknak ezek az előfordulásai más szinkronizációs egységhez nem tartozhatnak.

A szövegszinkronizációs módszerek feladata a szinkronizációs egységek és a köztük levő kapcsolatok meghatározása. Cél, hogy az egyes szinkronizációs egységek minél kevesebb szövegbeli egységből álljanak, és csak akkor tartalmazzanak több szövegbeli egységet, ha ez mindenképp szükséges (például két forrásszövegbeli mondatot egy fordításbeli mondattá vont össze a fordító).

1.2 A szinkronizáció típusai

Attól függően, hogy a szinkronizációs egységek milyen szövegegységekből állnak, megkülönböztetünk bekezdésszintű, mondatszintű, kifejezésszintű illetve szószintű szinkronizációt. A szövegek lefedettsége alapján megkülönböztetünk teljes és részleges szinkronizációt. A szinkronizációt teljesnek tekintjük, ha a forrásszöveg és a fordítás valamennyi egymáshoz rendelhető szinkronizációs egységét meghatároztuk; részleges szinkronizáció esetén a szinkronizációsegység-párok nem fedik le a forrásszöveg és fordítás valamennyi lefedhető szövegegységét.

A szinkronizációs módszerek ezen kívül különbözhetnek az alkalmazott technológiákban, pontosságban lefedettségben és robusztusságban. A következőkben két alapvető stratégiát mutatunk be, a szövegegységek hosszán alapuló módszert, amely teljes szinkronizációt tesz lehetővé, illetve a horgonyokat használó részleges szinkronizációt megcélzó módszert.

2. Szövegegységek hosszán alapuló módszer

A legtöbb mondat- és bekezdésszinkronizáló módszer a szinkronizálandó mondatok vagy bekezdések valamilyen mérték szerinti hosszán, illetve a Gale és Church által publikált módszeren [2][3] alapul.

A szinkronizálásnál az egyes szövegegységek hosszát mérhetjük karakterekben, szavakban vagy például a jelentés átvitele szempontjából legfontosabb szófajú szavak számát meghatározva. A betűírást használó nyelvek esetében a karakterszám a másik két lehetőségnél pontosabb összehasonlítást tesz lehetővé [1][2]. A jelentés átvitele szempontjából fontos szófajú szavak számát összehasonlító módszerekre a nem betűírást használó nyelveknél van szükség.

Gale és Church a szövegegységek hosszainak valószínűségi modellen alapuló összehasonlítását választotta [2][3]. Az összehasonlítás azon a feltételezésen alapul, hogy amennyiben L1 és L2 nyelvű szövegek egymás fordításai, és az L1 nyelvű szöveg minden egyes karaktere véletlen számú L2-beli karakter megjelenését okozza,

akkor ezek az egyes karakterekhez rendelt valószínűségi változók függetlenek és mind azonos normál eloszlással jellemezhetők.

Mondat- illetve bekezdésszinkronizációs módszerek esetében meg kell határozni, hogy milyen szinkronizációsegység-pár típusokat kezelnek a módszerek. Legegyszerűbb esetben a fordító egy forrásnyelvi mondatot egy mondatban fordít le, ilyenkor a párosított szinkronizációs egységek közül a forrásszöveghez és a fordításhoz tartozó is egy-egy mondatot tartalmaz (1-1 megfeleltetés). Emellett célszerű kezelni a 0-1 (beszúrás), 1-0 (elhagyás) 1-2 (részekre bontás), 2-1 (összevonás) és a 2-2 (mondathatár eltolás) típusú megfeleltetéseket.

Gale és Church a mondatok szinkronizálására egy dinamikus programozásra épülő algoritmust javasolt. A dinamikus programozás olyan globális optimumkereső eljárás, amely a feladat optimális megoldását alkalmas részfeladatok megoldásával éri el. Mondatszinkronizáció esetében az egyes szinkronizációsegység-összerendelésekhez költségeket rendelünk, majd a költségek összegét akarjuk minimalizálni. A költségek Gale és Church módszere esetében az egymáshoz rendelt szövegrészek hosszaránya alpján számított költségből és az alkalmazott szinkronizációs kategóriához (összerendelés típushoz) rendelt költségből állnak össze.

A dinamikus programozás előnye, hogy biztosan megtalálja a legkisebb költségű teljes szinkronizációt, és hogy nem használ véletlent, így bármikor alkalmazzuk ugyanarra az eredményre jutunk. A módszer hátránya a viszonylag nagy számítási- és memóriaigénye: M szövegegységnek N szövegegységgel történő szinkronizációjakor (M+1)×(N+1)-es táblázatot kell memóriában tartani és kitölteni. A táblázatnak azonban elég csak az átló körüli elemeit kiszámítani [1] így lényegesen gyorsabb változata készíthető az algoritmusnak, amely nem globálisan optimális megoldást keres már, hiszen csak a várhatóan legjobb megoldásváltozatok közül választja ki a legjobbat.

Gale és Church módszere az elhagyott és beszúrt szövegegységek kezelésekor nem bizonyult elég pontosnak, ami nem csoda, hiszen a szövegeket pusztán különböző hosszúságú szövegegységek sorozataként tekinti: a szövegegységek hosszán kívül más információt nem használ fel. Egyszerű (sok 1-1 megfeleltetést tartalmazó) szövegpárok esetében a módszerrel 95% feletti pontosságot értek el, azonban nehezebb szövegek esetében a módszer nem működött ilyen jól [5][6].

3. Horgonykeresés

Szövegpárok részleges szinkronizálása érdekében a szövegek pontjai között egy-egyértelmű megfeleltetést teremtve horgonynak (anchor) nevezett kapcsolatok kereshetők. Horgonyt alkothat bármilyen szövegegység-pár, ha a két szövegben egymás megfelelőinek tekinthetők.

3.1 A lehetséges horgonyok kiválasztása

Horgonyként, azonos karakterkészlettel írt nyelvek esetében választhatók a két szövegben előforduló azonos alakú szavak (homograph). Simard, Foster és Isabelle

annak érdekében, hogy minél több horgonyt találjanak, a hasonló alakú szavakat (cognate) is alkalmasnak találták a két szöveg közti kapcsolatok felvételére [5].

A különböző horgonytípusok közti választás során a legfontosabb szempontként a gyakoriságukat, illetve feltételezhető megbízhatóságukat érdemes megvizsgálni. A megbízhatóság érdekében a hibásan felvett párokat valamilyen módszerrel érdemes szűrni. A horgonykeresés az agglutináló és flektáló nyelveknél nehézségekbe ütközhet, hiszen a szóalakok a toldalékolásnak, illetve hajlításnak megfelelően változnak.

3.2 Hibás horgonyok szűrése statisztikai módszerekkel

A hibásan felvett horgonyok szűrésére eddig a legmegbízhatóbbnak tekinthető módszert Ribeiro, Lopes és Mexia publikálta [4]. A korábban ismertetett heurisztikus módszerekkel ellentétben Ribeiro és társai két statisztikai szűrőt definiáltak: mindkettőt a horgonyjelöltek szövegbeli pozíciói alapján kiszámítható lineáris regressziós sáv alkalmazásával. Első lépésben a lineáris regressziós sáv körül egy adaptív hisztogram alapú szűrővel meghatározott tartományon kívül eső pontokat vetették el, majd a regressziós sáv konfidenciasávján kívüli pontokkal tették ugyanezt.

4. Hibrid szinkronizációs megoldás

A szövegegységek hosszán alapuló módszer előnye, hogy elég robusztus, nem függ attól, hogy találhatók-e megfelelően pontos horgonyok a szövegpárban, ugyanakkor a beszúrásokat, elhagyásokat rosszul kezeli a módszer. Horgonyok használatával pontos, de csak részleges szinkronizáció érhető el. A horgonyok által elért részleges szinkronizáció sajnos nem alkalmas a szöveg kisebb szinkronizálandó szegmensekre bontására [1], így a két módszert csak egyetlen szinkronizáló algoritmusba integrálva lehet ötvözni. A következőkben az általunk választott hibrid, horgonyokon és szövegegységhosszakon alapuló megoldást mutatjuk be.

4. 1 Horgonyok keresése és szűrése

Legmegbízhatóbb horgonynak a szövegekben azonos számban előforduló, nagybetűket vagy számjegyeket tartalmazó szavakat választottuk. A számok esetében célszerűnek találtuk a tizedespontok, tizedesvesszők, és az esetlegesen a számjegyek között előforduló szóközök törlését az összehasonlítás előtt, így (a latin betűs nyelvek esetében) nyelvfüggetlen számformátumot hozva létre. A számjegyek között megengedtünk egyéb karaktereket is, így a termékkódokat, telefonszámokat és egyéb számokat tartalmazó szavak is horgonypontokká válhattak.

A magyar toldalékolás következtében az egyes szavak szóalakja változhat, ez ellen kétféleképp lehet tenni: morfológia alkalmazásával vagy Simardhoz és társaihoz hasonlóan az azonosan kezdődő szavak keresésével. Az ismeretlen szavak kezelésére is alkalmas morfológia alkalmazásával pontosabb megoldás készíthető, így a szavak morfológiai rendszerekkel történő tövesítése mellett döntöttünk.

A horgonyok szűrésére Ribeiro, Lopes és Mexia statisztikai alapú megoldásait [4] választottuk, amelyeket az előzőekben már röviden ismertettünk.

4.2 Hibrid algoritmus

A Gale és Church által kidolgozott szöveghosszakon alapuló algoritmust [2][3] vettük alapul, azonban a dinamikus programozás során nem csak a szövegegységek hosszának arányán és a választott szinkronizációs kategórián alapuló szinkronizációs költséget használjuk fel, hanem – előre meghatározott súllyal figyelembe véve – a horgonyok alapján számított (a költségekkel ellentétes előjelű) hasznot is.

A horgonyok alapján meghatározott haszon kiszámítására egy olyan egyszerű, heurisztikus módszert alkalmazunk, amely a fordítás során történő beszúrások és elhagyások felismerését elősegíti. A heurisztika alkalmazása elkerülhetetlen, mivel nincsen tudományos alapokon nyugvó elmélet, amely a feladat megoldása során alkalmazható lenne. A szövegegység-hosszakon alapuló módszerrel való kombinálhatósághoz egy valószínűségi alapokon nyugvó horgonyelőfordulás modell lenne szükséges, ilyet azonban már csak azért sem alkothatunk, mert a horgonyok várható eloszlását nem ismerjük. (A horgonyok eloszlása nem tekinthető egyenletesnek, a szövegekben csomósodást mutatva néhol egyszerre több horgony szerepel egy helyen; máskor ritkábbak, néhol pedig teljesen hiányoznak a szövegből a horgonynak tekinthető párok. Annak ismeretében, hogy a szövegek egyes részei más és más szerepet töltenek be, ez a jelenség nem tekinthető különösnek, a szövegekről alkotott képünkkel egybevág.)

A heurisztikus megoldás szükségszerűségének rövid indoklása után kezdjük rögtön a legfontosabbal, a horgonyok alapján számított haszon kiszámításának javasolt módjával. A hasznot a következő heurisztikus képlettel számítjuk:

$$haszon = \frac{\dfrac{\textit{a résztvevő szövegegységek közös horgonyainak száma}}{\textit{a résztvevő szövegegységek összes horgonyának száma}}}{\textit{résztvevő szövegegységek száma}} \qquad (1)$$

Az (1) képletet akkor alkalmazzuk, ha vannak horgonyaink (ha nincsenek, akkor értelemszerűen csak a szövegegységhosszak alapján lehetséges a szinkronizációs költség meghatározása). A képlet fő törtjének számlálójában levő törtkifejezés azt részesíti előnyben, ha az adott szinkronizációs kategória által meghatározott szövegegységekhez tartozó horgonyok nagy része a szinkronizációs kategória által meghatározott szövegegységeken belüli szövegpontokat köt össze (ezáltal szükség esetén az összevonásokat preferálva, ha ezt közös horgonyok indokolják). A nevező ezzel szemben a felesleges összevonásokat próbálja elkerülni, azáltal, hogy kisebb hasznot rendel a több szövegegységet összevonó szinkronizációs kategóriákhoz.

A fenti heurisztika nem preferálja, ha egy szinkronizációsegység-pár több horgonnyal is össze van kapcsolva: azt részesíti csak előnyben, ha a horgonyok közül minél több a szinkronizációsegység-páron belüli pontokat köt össze. Bonyolultabbá tenné a módszert, ha az összekötő horgonyok abszolút számát is fel akarnánk használni, hiszen ekkor a résztvevő szövegegységek hosszát is számításba kellene vennünk, mivel hosszabb szövegegységen belül nagyobb valószínűséggel találunk (több) horgonyt. A horgonyok nem egyenletes eloszlása miatt azonban nem tartottuk

szerencsésnek a szöveghosszal való összevetést. Bár a módszer feltehetően működne, az első kísérletekhez megfelelőbbnek találtuk egy egyszerű és átlátható heurisztika választását (amit az eredmények, – úgy tűnik –, utólag igazolnak is).

Az (1) képlet törtjének nevezőjébe akkor is beszámítjuk a résztvevő szövegegységet, ha az adott szövegegység nem is tartalmaz horgonypontot (csak egy vagy több másik, az adott szinkronizációsegység-pár részét képező szövegegység). Ez az eljárás megkérdőjelezhető, későbbi kísérletekkel kell majd eldönteni, hogy érdemes-e így eljárni. A szövegegység beszámításának előnye, hogy a beszúrt (vagy elhagyott) szövegegységek esetében nem támogatja a beszúrt (vagy elhagyott) szövegegység hozzávételét egy másik szövegegységpárhoz. Hátránya lehet, hogy akkor sem támogatja a 2-1 vagy 1-2 típusú összerendeléseket, ha azok szükségesek. Megoldás lehet, ha a horgonyokon alapuló heurisztikusan számított haszon számításba vételét meghatározó súlyt úgy határozzuk meg, hogy amikor a szövegegység hosszakon alapuló módszer az adott összerendelést nagymértékben támogatja, akkor az összerendelés létrejöhessen.

4.3 Eredmények

Az előzőekben ismertetett hibrid módszert olyan angol illetve magyar nyelvű informatikai témájú szövegeken teszteltük, amelyek a csak szövegegységhosszakon alapuló módszerrel a beszúrások és elhagyások miatt gyakorlatilag szinkronizálhatatlanok voltak. A problémás helyek közelében a hibrid módszer a horgonyoknak köszönhetően többnyire helyesen határozta meg a szinkronizációs egységeket. Horgonyok hiányában természetesen a csak szöveghosszakon alapuló módszerrel azonos eredményt ért el a hibrid megoldás is, ezért a későbbiekben a megbízható horgonyok számának növelésével a már most is megfelelő eredmények tovább javíthatók majd.

Referenciák

1. Pohl Gábor: Fordítások terminológiai konzisztenciájának vizsgálata (diplomaterv feladat). Budapesti Műszaki és Gazdaságtudományi Egyetem (2003)
2. Gale, William A.; Kenneth W. Church: A Program for Aligning Sentences in Bilingual Corpora. In: 29th Annual Meeting of the Association for ational Linguistics (1991)
3. Gale, William A.; Kenneth W. Church: A Program for Aligning Sentences in Bilingual Corpora. Computational Linguistics, Volume 19, Number 1, March 1993, Special Issue on Using Large Corpora: I.
4. Ribeiro, António; Gabriel Lopes; João Mexia: Using Confidence Bands for Parallel Texts Alignment. In: Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics (2000).
5. Simard, Michael; Foster, George; Isabelle, Pierre: Using Cognates to Align Sentences in Bilingual Corpora. In: Proceedings of TMI-92, Montréal, Canada. (1992) pp. 67-81
6. Langlais, Philippe; Michel Simard; Jean Véronis: Methods and Practical Issues in Evaluating Alignment Techniques. In: 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics (1998)

Text Alignment Methods, Hybrid Paragraph and Sentence Alignment Technique

Gábor Pohl

Péter Pázmány Catholic University, Budapest
Department of Information Technology
pohl@morphologic.hu

Keywords: text alignment, paragraph alignment, sentence alignment, statistically filtered anchors, hybrid alignment technique

Abstract

This paper contains an introduction to text alignment methods and presents a new state of the art hybrid text alignment technique.

In the introductory part, alignment is defined in a generic way as a one-to-one correspondence of alignment units, where an alignment unit is a set of text units (paragraphs, sentences, etc.) that are not contained in other alignment units. After the explanation of the definition two different alignment strategies—the strategy of character length based complete alignment and the strategy of anchor based partial alignment—are described.

After the introduction to alignment techniques a new hybrid method is presented. In order to achieve high precision alignment of text pairs (with omitted and inserted text units) the char-length based method is combined with the use of statistically filtered anchors. In order to find the same anchors in both texts, anchor candidates are also considered in their lemmatized forms. The methods described by Ribeiro et al. have been used to filter anchor candidates. Integrating the two alignment strategies in one alignment framework, we present a new heuristic formula that defines the *revenue of anchors* that can be used in the dynamic programming algorithm used by the length based method.

Nyelvi tudásra épülő fordítómemória

Hodász Gábor[1], Gröbler Tamás[2]

[1]Pázmány Péter Katolikus Egyetem
Információs Technológiai Kar
Budapest
hodasz@morphologic.hu
[2]MorphoLogic Kft.
Budapest
grobler@morphologic.hu

Kivonat. A cikkben bemutatásra kerülő MetaMorpho TM rendszer olyan fordítástámogató eszköz, amely a hagyományos fordítómemória-funkciókat nyelvi intelligenciával kiegészítve a jelenlegi rendszereknél többször ajánl fordítást, és azok jobban közelítik a kívánt minőségű fordítást. A fordítás egységei a mondatnál kisebb szegmensek (főnévi szerkezetek és az ezeket tartalmazó mondatvázak), amelyeket a forrás- és célnyelvi elemzők állítanak elő. Az adott bemeneti mondathoz hasonló szegmenseket „nyelvi intelligencián" alapuló távolság segítségével keressük, és a megszülető új fordításokat mint szabályokat tároljuk, amelyek a gépi fordítás minőségét folyamatosan javítják.

Bevezetés

Napjainkban a leginkább elterjedt fordítástámogató eszközök a fordítómemóriák (translation memory, TM), amelyek a fordítási munkák során keletkező párhuzamos szegmensek eltárolásával nyújtanak segítséget a fordítónak. A hagyományos fordítómemóriában a szegmensek a teljes mondatok, amelyek nyelvi tudás nélkül kerülnek tárolásra, és közülük karakter-alapú távolság alapján választ a rendszer az aktuális mondathoz hasonlót, majd ez alapján ajánl fordítást a felhasználónak. Bár a legtöbb rendszer felismeri és kezeli a nem fordítandó és a szabadon behelyettesíthető elemeket (számokat, dátumokat stb.), valamint rendelkezik terminológia-kezeléssel, azonban nem képesek kezelni a pusztán morfológiai különbségeket, vagy a szintaktikailag hasonló, de karakteresen különböző mondatokat.

A szabály alapú gépi fordító rendszerek (rule based machine translation, RBMT) a fordítómemóriákkal szemben a fordítási folyamat emberi beavatkozás nélküli automatizálását célozzák meg. Nyelvi tudással rendelkeznek, elemző algoritmusok és fordítási szabályok segítségével végzik a fordítást, általában kötött nyelvpárokra. A nyelvi kétértelműségek és az elemző algoritmusok tökéletlensége mellett a szabálybázis korlátozott bővíthetősége is gátat szab ezen rendszerek pontosságának.

Nagao a '80-as években új megközelítést javasolt a szabály alapú fordítás hibáinak megoldására: a példa alapú fordítást (Example Based Machine Translation, EBMT) [1]. Az ötlet alapja az a pszicho-lingvisztikai megfigyelés volt, hogy a fordítási

folyamat során az emberi fordító is használja a szónal nagyobb, de mondatnál kisebb szerkezeteket. A megfigyelések szerint minél tapasztaltabb a fordító, annál nagyobb egységeket használ [2]. Az EBMT rendszerekben a fordítás alapjai a szónál nagyobb, de mondatnál kisebb nyelvtani egységek, amelyeket párhuzamos korpuszból állít elő a rendszer [3]. A fordítási folyamat során az egyes visszakeresett mintákból szabályok állítják össze a célnyelvi mondatot. A kutatások célja, hogy megtalálják az optimális utat a két szélsőnek tekinthető megközelítés között. Több szerző kimutatta, hogy a fordítás minősége függ a rendszer nyelvi tudásának mennyiségétől [4]. Így az alkalmazott nyelvtechnológiai algoritmusok pontosságának és hatékonyságának növelése szintén tárgya a jelen kutatásoknak.

A cikkben bemutatásra kerülő MetaMorpho TM rendszer szintén a két alapvető iránynak az ötvözését tűzi ki célul, így az EBMT rendszerek közé sorolható [5].

A MetaMorpho TM működésének vázlata

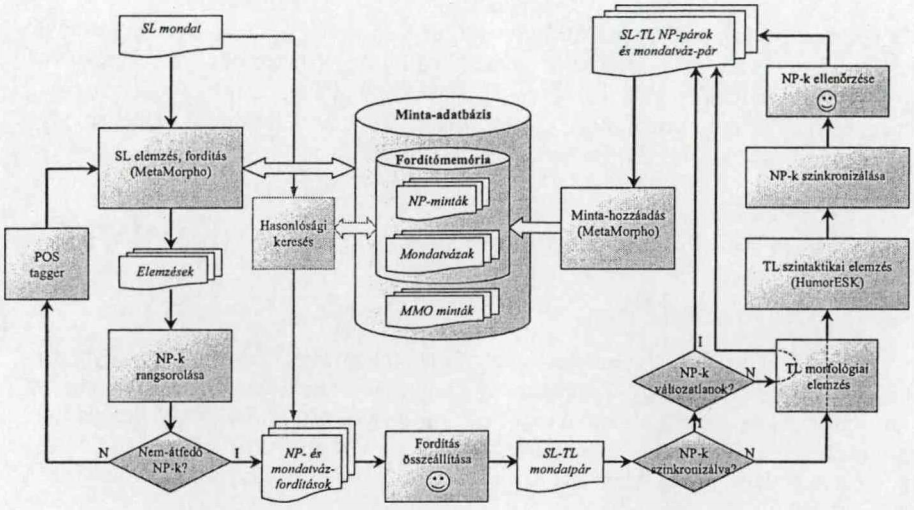
A MetaMorpho TM alapvetően fordítómemóriaként működik, azaz fejlett eszközökkel támogatja az emberi fordítót, valamint lehetőséget ad az adatbázis bővítésére. A fordítói munka során a fordított mondatok feldolgozásával bővül a szabálybázis, és emellett lehetőség van minták párhuzamos korpuszból való automatikus felvételére is.

A fordítás előállítása

A fordítási munka során az emberi fordító felügyeli a fordítás folyamatát: módosítja vagy elfogadja a rendszer által felkínált fordításokat.

A folyamat lépései a következők (1. ábra):

- A beérkező forrásnyelvi mondatot a morfoszintaktikai elemző lemmákra bontja.
- A főnévi csoportok (NP-k) és a többi lemmából álló mondatváz alapján az „intelligens" kereső hasonló mondatot, illetve hasonló NP-ket keres az adatbázisban.
- A találatokat megfelelően szűrve és rangsorolva előáll a célnyelvi szegmensfordítás-jelöltek listája.
- A jelöltekből, illetve az opcionálisan gépi fordítással előállított célnyelvi szegmensekből összeáll az eredeti mondat felajánlott fordítása.
- A felajánlott fordítás(oka)t a felhasználó elfogadhatja vagy módosíthatja.
- A módosított célnyelvi szegmensek elemzése és forrásnyelvi párjukkal való eltárolásuk révén új szabályokkal bővül a fordítómemória.



1. ábra:A MetaMorpho TM működése

A fordítómemória bővítése

A fordítómemória bővítése során új szabályokat adhatunk hozzá a szabálybázishoz.

A fordítás folyamán az emberi fordító által javított célnyelvi mondatok elemzése és a szegmens forrásnyelvi párjával való összerendelése révén olyan párhuzamos korpuszelem jön létre, amelyből előállítható a megfelelő formában a fordítási szabály. A rendszer jelenlegi verziójában a mondatpárokból kiemeljük a főnévi csoportokat, amelyekből önálló szabály-minta keletkezik. Ugyancsak szabály-minta keletkezik a mondat fennmaradó részéből, az ún. mondatvázból, amelyben az NP-k helyét üres hely jelzi. A későbbi fordítások során ezek a helyek más NP-kkel is kitöltődhetnek, amennyiben azok kielégítik a megfelelő feltételeket. Azért, hogy az eredeti mondat egyértelműen visszaállítható legyen, a mondatvázból születő szabályban eltárolásra kerül az eredeti NP-k azonosítója is.

Párhuzamos korpuszból való automatikus szabályfelvétel esetén a fentebb vázolt folyamat emberi beavatkozás nélkül megy végbe. A külön modulként kifejlesztendő mondatszinkronizáló (aligner) által előállított párhuzamos mondatokból a nyelvi elemző előállítja az NP-ket és a mondatvázat. Az így létrejövő mintákat mint szabályokat hozzáadjuk a szabálybázishoz.

A szabály alapú fordító néhány jellemzője

A MetaMorpho TM intelligens fordítómemória a MorphoLogic MetaMorpho nevű szabály-alapú fordítórendszerére [6] támaszkodik. A MetaMorpho rendszerben a forrásnyelvi mondat elemzésének egyes lépéseivel egy időben párhuzamosan létrejön a megfelelő célnyelvi struktúra is. Így egy MetaMorpho szabály minden esteben egy forrás (angol) és egy célnyelvi (magyar) részből áll. Példa egy szabályra:

```
*NX=approach+to:12
EN.NX[ct=COUNT] = N(lex="approach") + PPOBJ(lex="to")
HU.NX = PPOBJ[case=GEN] + N[lex="megközelítés"]

;example: This is a really nice approach to religion.
```

A MetaMorpho szabályrendszerének másik jellemzője, hogy a szabálybázis homogén: nem különböztetjük meg a szótár-szerű és a szintaxis-szerű szabályokat.

A fenti két tulajdonság lehetővé teszi, hogy a fordítómemóriába kerülő szabályok bármilyen szintű nyelvtani struktúrát írjanak le, legyen az egyetlen főnév és fordítása, vagy egy mondatváz, amelyben üres helyek jelzik a főnevek helyét és a vonatkozó megkötéseket.

A szabályok az elemzés-fordítás folyamán egyszerű unifikációs nyelvtan szerint működnek. Az egyes szabályokban a különböző jegyek (megkötések) határozzák meg a szabály specifikusságát. A szabályban a nem kitöltött megkötések a konkrét mondat elemzése során kerülnek kitöltésre. Így a fordítómemória működése folyamán egy korábban eltárolt minta akkor is releváns lehet az aktuális mondat fordításában, ha előzőleg más morfológiai jegyekkel szerepelt. Ehhez az szükséges, hogy a szabályok eltárolásakor meghatározzuk a kellő megszorításokat, a többi jegyet azonban kitöltetlenül hagyjuk. Így a fordítómemória által megtalált korábbi fordítás csak abban az esetben lesz jelölt, ha kielégíti a szükséges megszorításokat. A nem szükséges megszorítások (pl. szám, személy, idő stb.) pedig a célnyelvi mondat generálása során az aktuális forrásnyelvi megfelelőik szerint kerülnek kitöltésre. Ez a megközelítés lehetővé teszi, hogy a hagyományos fordítómemóriával szemben például az angol 'go' igének különböző idejű alakjait (pl. 'went', 'has gone' stb.) annak ellenére megtalálja a rendszer a minták között, hogy közöttük a karakter-alapú távolság igen nagy.

A felhasználónak felajánlott fordítások a memória által visszaadott hasonló minták (főnévi szerkezetek és mondatvázak) összeillesztésével és utófeldolgozásával állnak elő. Amennyiben a fordítandó mondat nem minden eleme található meg a memóriában, úgy a felhasználó opcionálisan kérheti a fenmaradó részek gépi fordítását.

Minták és szabályok

A MetaMorpho TM rendszer átmenetet képez a minta (memória) alapú és a szabály alapú fordítástámogató illetve fordító eszközök között. A fentebb leírt tulajdonságai miatt egyesíti a minta és a szabály fogalmát. A fordítómemória egy eleme egyaránt

párhuzamos korpuszelem és fordítási szabály, és a felépített memória egyaránt tekinthető párhuzamos annotált korpusznak, valamint szabálybázisnak. Egy feltöltött memória alkalmas lehet korpusznyelvészeti célokra, például terminológia-keresésre, glosszáriumépítésre stb. Ugyanakkor a szabályok a szabályalapú fordítás minőségét javíthatják.

Hasonlósági keresés

A korábban eltárolt minták közötti keresés fontos lépése a fordítómemória működésének. A hagyományos, ma kereskedelmi forgalomban kapható fordítómemóriák az egyes minták közötti karakter-alapú távolság alapján keresnek az adatbázisban. A MetaMorpho TM jövőben kifejlesztendő, nyelvi intelligenciával ellátott kereső algoritmusa egyaránt képes felismerni az azonos lexikai elem más morfológiai alakjait, valamint a kissé különböző szerkezetű szintaktikai egységeket is.

A dinamikus programozáson alapuló algoritmus három szinten vizsgálja a szegmensek hasonlóságát:

1. felszíni alak
2. morfológiai jegyek
3. szófaj

Az egyes szinteken a különböző típusú különbségek különböző büntető pontokat eredményeznek. Az összehasonlítást „alulról felfelé" végezzük, azaz amennyiben a *n*-dik szinten egyezést találunk, tovább vizsgáljuk az *(n-1)*-dik szinten. Amennyiben a két szegmens nem azonos számú lemmából áll, úgy a büntetés függ attól, hogy a hiányzó vagy más értékű lemma az adott szerkezet feje vagy más eleme.

Várakozásaink szerint ez a hasonlósági keresés és az ennek megfelelően indexelt adatbázis nagyobb számban fog megfelelő fordítás-javaslatot adni, és a javaslatok jobban közelítik a kívánt fordítást.

Implementáció

A fentiekben felvázolt MetaMorpho TM intelligens fordítómemóriát első lépésben angol-magyar nyelvpárra valósítjuk meg. A rendszer moduljait C++ nyelven implementáljuk. Az egyes modulok megvalósítják a különböző elemző illetve generáló funkciókat: mondatszegmentálás, szószegmentálás, morfológiai elemzés stb. A modulok közös adatstruktúrát dolgoznak fel, és bármely feldolgozási lépés eredménye XML formátumban is megjeleníthető.

A szintaktikai szabályokat a [7]-ben leírt modell szerint valósítjuk meg.

A fordítómemória funkciót relációs adatbázis valósítja meg. Az egyes táblákban tároljuk a szabályokat valamint a közöttük lévő leszármazási kapcsolatokat. A hasonlósági keresés és index jelenleg még implementálásra vár.

Összefoglalás és további munkák

Jelen munkánkban bemutattuk a MetaMorpho TM intelligens fordítómemóriát, mely az irodalomban leírt EBMT elv szerint a szónál nagyobb, a mondatnál kisebb nyelvi egységek feldolgozásával, tárolásával és összeillesztésével nyújt hatékony segítséget az emberi fordító számára. A morfológiailag elemzett főnévi szerkezetek, valamint az ezeket üres helyekként tartalmazó mondatvázak eltárolásával és nyelvi távolságon alapuló hasonlósági keresésével a hagyományos fordítómemóriáknál várhatóan több korábbi fordítást tud felajánlani a rendszer, valamint az egyes fordítási egységek összeillesztésével a kapott fordítás jobban közelíti a kívántat.

Az elvégzendő további munkák között szerepel a mintákat tároló relációs adatbázis továbbfejlesztése, a hasonlósági keresés implementálása, valamint a célnyelvi oldalon az egyes fordítási egységek összeállítása helyes mondatokká. Szükséges továbbá a felhasználó által megadott célnyelvi mondat nyelvi elemzésének megvalósítása, amely lehetővé teszi a memória bővítését, előállítva a párhuzamos elemzett mintákat (szabályokat). A nyelvi motort a fordítómemória-rendszerekhez hasonló integrált fordítástámogató szoftverré kívánjuk fejleszteni.

Irodalom

1. Nagao, M, 'A framework of a mechanical translation between Japanese and English by analogy principle', In A. Elithorn and R. Banerji (eds.) (1984), Artificial and human intelligence, 173-180. Amsterdam: North-Holland.
2. Gerloff, P. 'Identifying the Unit of Analysis in Translation', in Færch & Kasper (eds.) (1987) *Introspection in Second Language Research*, Clevedon: Multilingual Matters, pp. 135-158.
3. Turcato, D. & F. Popowich, 'What is Example-Based MT?' *Proceedings of the Workshop on Example-Based Machine Translation.* (2001) http://www.eamt.org/summitVIII/workshop-papers.html
4. McTait, K., 'Linguistic Knowledge and Complexity in an EBMT System Based on Translation Patterns'. (2001) *Proceedings of the Workshop on Example-Based Machine Translation.* http://www.eamt.org/summit VIII/workshop-papers.html
5. Prószéky, G. and L. Tihanyi, 'MetaMorpho: A Pattern-Based Machine Translation Project'. (2002) *Translating and the Computer 24,* ASLIB, London.
6. Hegedűs, B. 'Természetes nyelvű információk számítógépes feldolgozása', (2003) Diplomaterv, Budapesti Műszaki és Gazdaságtudományi Egyetem, Számításelméleti és Információtechnológiai Tanszék, Budapest.
7. Prószéky 'Syntax As Meta-morphology', (1996) *Proceedings of COLING-96,* Vol.2, 1123–1126. Copenhagen, Denmark.

A Linguistically Enriched Translation Memory

Gábor Hodász

Faculty of Information Science
Pázmány Péter Catholic University
Budapest

hodasz@morphologic.hu

Tamás Grőbler

MorphoLogic Ltd.

Budapest

grobler@morphologic.hu

Keywords: translation memory (TM), machine translation (MT), computer-aided translation (CAT)

Translation memories (TMs) offer translations based on previously stored translation units. Traditionally, translation units are sentence pairs, and the search method is based on character-level similarity. Most existing systems recognise "do not translate" elements such as dates or numbers and are equipped with terminology databases. Current TM systems, however, fail to handle morphological differences or syntactically similar sentences that are very different on the character level. We propose a TM system that applies the machinery of a rule-based machine translation (RBMT) system to compose the target sentence from the stored sub-sentential translation units.

MetaMorpho TM, a linguistically enriched TM system, uses MorphoLogic's fine-grained language technology in both languages to yield more translations that are also more exact matches to the source sentence. In the current version of the system, translation units are noun phrases (NPs) and the sentence skeletons that incorporate them. The stored NPs consist of morphologically analysed terminal symbols. Sentence skeletons contain the NP slots and the rest of the sentence as analysed terminal symbols. The engine is being integrated with a word processor to allow translators to select the best match and prepare the translation with minimal interaction.

Database lookups are performed by the MetaMorpho machine translation system. Translation is performed at the same time as syntactic parsing. When applied to the translation memory, MetaMorpho interprets translation units as highly lexicalised rule pairs and attempts to compose the target sentence from them. As the representation of the MT and the TM samples are compatible, our RBMT system also benefits from using the memory building mechanisms in-house.

The actual memory is implemented as a relational database. To further increase the number of matches, the database is also indexed using a "linguistic similarity" measure. Automatic addition of new items to the database is available both during translation and from a parallel corpus. The latter will be supported by the aligner software to be developed as a separate module.

In the paper, we show examples to demonstrate how sentences are translated, how new samples are added to the database, and how they are reused in another translation.

Új módszerek az emberi fordítás számítógépes támogatásában

Kis Balázs, Lengyel István

MorphoLogic Kft.
{kis,lengyel}@morphologic.hu

A piacon jelenleg élesen elkülönül a fordítástámogató eszközök és a gépi fordítóprogramok kategóriája. A szerzők megvizsgálják a fordítástámogató eszközök lehetőségeit, az elkülönülés okát, és javaslatot tesznek a két csoport közelítésére, a szinergia kiaknázására. A cikk ismerteti a szerzők elképzelését az ideális fordítástámogató csomagról: az intelligens fordítómemóriáról, amely a statisztikai hasonlóságkeresésen kívül számítógépes nyelvészeti eszközöket is felhasznál, a csoportmunkát támogató, a terminológiát rugalmasan kezelő terminológiakezelőről, a szöveg terminológiai előkészítését részben automatizáló terminuskeresőről és az egész rendszert egybefogó fordítási munkafolyamat-automatizálási rendszerről. Megvizsgálja annak előnyeit és hátrányait, hogy a fordítómemória üresen kerül a fordítóhoz, és foglalkozik a fordítómemória-adatbázisok fejlesztésének lehetőségével.

1. A számítógépes fordítástámogatás szükségessége

A fordítók munkájuk során számtalanszor kerülnek olyan helyzetbe, hogy rutinfeladatokat kell végrehajtaniuk. A gépi fordítással szemben a számítógépes fordítástámogatás nem az emberi intelligencia kiváltását, hanem annak kiegészítését, hatékonnyá tételét célozza meg. A fordítástámogatási szoftverek célja a rutinfeladatok automatizálása, ezáltal az egy fordítási egység lefordításával/célnyelven történő véglegesítésével[1] töltött átlagos idő csökkentése és a fordítás minőségének javítása. A fordítás minőségének értékelése a fordítástudományi szakirodalomban vitatott kérdés (Klaudy 2003), de abban minden szerző egyetért, hogy a terminológiai, stiláris stb. konzisztencia alapvető ismérve a jó fordításnak.

A jelenleg elérhető fordítástámogató (CAT – *Computer Assisted Translation*, számítógéppel támogatott fordítás) eszközök három kategóriába sorolhatók:

1. Fordítómemória: a fordító, illetve a fordítóközösség korábbi fordításainak újrahasznosítására;

[1] A fordítási egység célnyelven történő véglegesítése alatt a szerzők a szöveg előkészítésével kezdődő, a fordítást, lektorálást, esetleg korrektúrázást és olvasószerkesztést, nyomdai előkészítést magába foglaló folyamatot értik, amelybe beletartozik a szöveggel kapcsolatos projektmenedzsment is.

2. Terminológiakezelő rendszer: a fordítás témakörének megfelelő terminológia hatékony megkeresésére és szótárazására;
3. Munkaszervező eszköz: a csoportmunkában végzett fordítás szétosztására, összegyűjtésére, továbbítására, mérésére és egyéb szervezésére.

2. A fordítómemória

A fordítómemória működése azon a feltételezésen alapul, hogy a forrásnyelven íródott egyforma mintákat egyforma módon kell lefordítani a célnyelvre. Ez a feltételezés legtöbb esetben jogos, kivétel, amikor egy adott regiszter a forrás- vagy célnyelven nem létezik. Előnye, hogy a fordítócsoportok outputját is egységesíti, kollektív tudást hoz létre a meglévő fordítások hasznosítása révén. A terminológiakezelést támogató szoftverek azon a feltételezésen alapulnak, hogy vannak olyan kifejezések, amelyek egy adott nyelvről egy másik nyelvre egyértelműen fordíthatók az adott szövegkörnyezetben. Éppen ezért az ilyen szoftverek nem csupán a szócikket tartalmazzák, hanem annotáció révén meghatározható bennük az adott szócikk érvényességi tartománya – azon szövegek típusa, amelyekben az adott kifejezés terminusnak tekinthető. A munkaszervező eszköz a fordítók munkáját közvetlenül nem könnyíti: a fordításszervezők tapasztalatai alapján alakult ki, és az ő munkájuk minél szélesebb körű automatizálását tűzi ki célul.

A piaci forgalomban jelenleg kapható fordítástámogató programok fejlesztése piacvezérelt módon történik, amelynek lényege, hogy olyan terméket készítsenek, amely minél szélesebb réteg által használható. Az ilyen szoftverek éppen ezért nyelvfüggetlenek: így a termék potenciális vásárlói bázisa nem csak egy nyelv vagy nyelvpár fordítóira terjed ki. E megközelítés hátránya, hogy nyelvi elemzés nélkül a fordítómemória funkcióját (az aktuális forrásszöveg szegmenseivel megegyezők vagy hozzájuk hasonlók keresése a korábbi fordítások adatbázisában) csak részben tudja betölteni. A hasonlóságok keresése csak statisztikai módon történhet, amelybe bizonyos fokú intelligenciát a fuzzy logika visz, hiszen lehetővé teszi az alulspecifikált összehasonlításokat. A jelenleg kapható fordítómemóriák egyike sem lép túl a szöveg stringként történő kezelésén, a hasonlóságok keresése is string alapon történik, a morfológiai és grammatikai információ absztrakt kezelése nem jelenik meg. A legelterjedtebb nyelvek (angol, francia) esetében ez a megközelítés a nyelvi információ explicit megjelenése miatt jó hatásfokkal működik, de a ragozást használó nyelveknél nem: a *sajt* és a *hajt* között ugyanakkora a hasonlóság, mint az *írom* és az *írod* között – 1 karakter. Az előbbi nyelvek esetében a viszonylag kötött szórend miatt a szavak távolsága elég sok információt hordoz, míg a kevésbé kötött szórendet alkalmazó nyelvek esetében a szavak távolságát nem elég figyelni: például a „*Vettem egy zöld kerékpárt.*" alapján a nyelvi elemzést nem támogató fordítómemória nem képes javaslatot adni a „*Pisti vett tegnap a régi biciklije helyett egy nagy, rikítóan piros, váltós férfi kerékpárt.*" mondatra. Megfelelően nagy szótárak nélkül azonban a jól támogatott nyelvek esetében sem lehet felismerni például az idiomatikus helyzeteket, ezért szükség van az idiómák olyan szabályokként történő értelmezésére, amely felülbírálja a többi nyelvtani szabályt. Ha azonban egy idióma ragozott formában szerepel a mondatban, a hagyományos fordítómemóriák ismét csődöt mondanak.

A fenti példákból látható, hogy a szórendet szemantikai szerepben felhasználó és ragozási sorokat alkalmazó nyelvek esetében a hatékony hasonlóságkeresés csak morfológiai és bizonyos szintű grammatikai elemzés révén valósítható meg.

Felismertük azt a *tényt*, hogy az eredeti szegmenshez hasonló szegmenst már fordítottak a program segítségével. Most vagy megelégszünk annyival, hogy megjelenítjük a fordító számára a hasonló szövegre eltárolt fordítást, vagy hozzáigazítjuk azt a jelenlegi forrásszegmenshez: felruházzuk a célszegmenst azokkal a nyelvtani tulajdonságokkal, amelyek a forrásszegmensre jellemzőek voltak. Ha például a forrásszegmens felszólító módú és E/2-re vonatkozik, szükség esetén átalakítjuk a tárolt fordítást felszólító módra, E/2-re. Ha a fordítónak nem kell vesződnie az apró nyelvtani módosításokkal, időt takarítunk meg a számára.

A jelenleg kapható fordítómemóriák előnye és hátránya egyszerre az, hogy üres adatbázissal érkeznek a felhasználóhoz. Így minden fordítómemória tartalma szubjektív, a világnak azt a szegmensét tükrözi, amellyel a fordító a gyakorlata során eddig találkozott. Ennek egyaránt vannak előnyei és hátrányai.

Előny, mert:

- A fordító/megbízó fordításaiból „tanul" csupán, ezért a fordító számára a lehető legmegfelelőbb találatokat adja, a fordító stílusától nem tér el.
- Biztosítja a fordítások konzisztenciáját fordító szintjén.
- Lehetőséget ad az egyéniség kibontakozására.

Hátrány, mert:

- Sok időt vesz igénybe az adatbázis feltöltése, azaz a fordítómemória hasznossá válásának elérése.
- A fordító stílusát konzerválja – hiába tanul meg a fordító később szebben fordítani, a memóriából a régi fordításai jönnek elő.
- Rögzülnek a fordító félrefordításai, konzisztens félrefordítás lehetséges.
- Sok időbe kerül a régi fordítások forrás- és célszegmenseinek összepárosítása, az *alignment* (elrendezés) művelete.
- Nehezen hozható összhangba több, addig külön dolgozó fordító munkája és stílusa, ha mindannyian használtak korábban is saját fordítómemória-adatbázisokat.
- Nem garantálható az egy szakterületen kialakult fordítási normákhoz való alkalmazkodás.

A fenti összefoglalóból látható, hogy az előnyökhöz képest többségben vannak a hátrányok, ezért érdemes lenne a fordítómemóriákat eleve adatbázissal együtt adni.

Egyes nagy megbízók már ma is ellátják a fordítókat a fordítási megbízás kezdetén fordítómemória-adatbázissal, azonban ez még nem tekinthető gyakorlatnak, hiszen a megbízók általában nem kapják meg a befejezett fordításuk fordítómemória-adatbázisát, maguk pedig nem építenek ilyen adatbázist.

A fordítómemória-adatbázis (ami végső soron egy szinkronizált korpusz) kiadása és értékesítése általában szerzői jogi problémákba ütközik, de gondos előkészítéssel mégis lehetséges úgy összeállítani jó minőségű szövegeket, hogy azok ne legyenek ellentétesek senki érdekével, ugyanakkor reprezentálják az adott szakterületen kialakult, normaként elfogadott tudást.

3. Terminológiakezelés

A jelenleg szokásos terminológiakezelő rendszerek nagy hátránya, hogy szabványszerűen kezelik a terminológiát, vagyis a terminus technicusokat egyértelműnek tekintik. A szerzők fordítói és terminológusi tapasztalatai szerint azonban a terminológia legfőbb attribútuma nem az egyértelműség, hanem adott nyelvi tartalom témaspecifikus megformálása, illetve az általános nyelvhasználatban is előforduló szavaknak, kifejezéseknek az általános használattól eltérő jelentéssel (eltérő kontextusban, esetleg eltérő szintaxissal) való használata. A terminológia így sem nem feltétlenül nominális, és nem is egyértelmű (még egy tárgykörön belül sem): szociolingvisztikai tény, hogy adott tárgykör terminológiája minden nyelven önállóan fejlődik, sokszor a szabványosítási folyamatoktól függetlenül vagy éppen azok ellenére.

A terminológiakezelés esetében a számítógépes fordítástámogatás szempontjából a terminológia a szerzők által javasolt definíciója: *Terminológia mindaz, amelynek inkonzisztens fordítása a fordítás érthetőségét rontja.* Ez a definíció megengedi, hogy egy adott nyelven terminusnak minősülő kifejezés fordításait ne tekintsük minden esetben, minden nyelven terminológiának, azaz ne rontsuk a fordítás egészét olyan, az adott nyelven idegenül hangzó fordításokkal, amelyeket csak azért fordítunk következetesen, mert a forrásnyelvi szöveg e szempontból következetes. Megengedi két kultúra szaknyelvében vagy nyelvében a szemantikai háló eltéréseit. Az ilyen szempont figyelembe vételével megalkotott szöveg a nyelvi elemek egyértelmű leképezése helyett a kontextus leképezését, a forrásnyelvi, az adott kultúrát figyelembe vevő kontextus újbóli, célnyelvi létrehozását jelenti. Például az angolszász jogrend, az ún. common law kifejezéseinek terminológiaként történő magyarítása teljességgel értelmetlen, mivel az angolszász jogrend alapjaiban különbözik a magyartól, és az egyes kifejezések használata – főleg, ha azok a jelenleg a magyar jogban használt kifejezések új jelentéssel való felruházása, angol terminusokkal történő megfeleltetése – azt a téveszmét keltené a magyar olvasóban, hogy az angol jogrendnek sok közös pontja van a magyarral. Felhozhatnánk még azt a példát is, hogy a tengerhajózásnak a tengeri nagyhatalmak nyelveiben sokkal kiterjedtebb terminológiája van, mint a magyarnak, egész egyszerűen az ország földrajzi körülményei miatt, vagy azt, hogy a számviteli beszámolók jó fordítása (azaz olyan fordítás, amely más számviteli környezetben – országban – élő emberek számára is egyértelmű), elképzelhetetlen a számvitel ismerete nélkül.

A terminológiát nyelvpárokra bontva kezelni hatékonyabb, mint többnyelvű terminológia esetében feltételezni, hogy egy kifejezést minden nyelven terminológiaként kell kezelni. A terminológia mind szűk, mind tág értelemben kontextusfüggő: szűk értelemben a szöveghez illeszkedik, tág értelemben pedig a célnyelvi kultúrához és a szöveg fogadójához, annak ismereteihez, tudásához. Mindezt figyelembe kell venni a terminológiakezelés során, ha a profi fordítók igényeit is kielégítő fordítástámogató eszközt kívánunk fejleszteni.

A terminológia megalkotása jó esetben csoportmunka révén alakul ki, ezért fontos, hogy a számítógépes terminológiakezelő eszköz képes legyen terminológiai fórumként is működni. A jelenlegi terminológiakezelők nem képesek státusokat megkülönböztetni egy adott terminusra. Megfelelően kifinomult jogosultságkezeléssel a fordítási folyamat minden résztvevője beleszólhat, javaslatokat tehet a terminusok kialakítására – például jelöljük 1-gyel azokat a fordításokat, amelyeket a fordító javasol, 2-vel

azokat, amelyeket egy másik fordító is elfogad, 3-mal azokat, amelyeket egy nyelvi lektor, 4-gyel azokat, amelyeket egy szaklektor, 5-tel azokat, amelyeket egy szakma több képviselője is elfogad. Az Európai Unió fordítási intézményeiben ugyan megoldották a terminológiaalkotás folyamatának szabályozását, de intézményközi megállapodás nincs, ezért mind a mai napig előfordul, hogy pl. az Európai Parlament és az Európai Bizottság két külön kifejezést használ olaszul egy francia kifejezésre. Fontosnak tartjuk egy olyan terminológiakezelő kifejlesztését, amelyben nem csak a végleges terminológia tárolása oldható meg, hanem a terminológiai javaslattétel és a viták is a rendszeren belül bonyolíthatók le.

A terminológiakezelő és a fordítómemória egyesítése szintén fontos kérdés. A piacon kapható terminológiakezelők ugyan együttműködnek a fordítómemóriákkal (általában a rendszerek mindkét alkalmazást tartalmazzák), de ezek sem alkalmaznak morfológiai elemzést, így nem képesek például a ragozott szavak felismerésére, csak akkor, ha azok külön szótári bejegyzésként vannak eltárolva.

4. Munkaszervezés

A munkaszervező (projektmenedzsment) eszköz ugyan szűk értelemben nem tekinthető nyelvtechnológiai eszköznek, de mivel a fordításnak vagy a fordítás véglegesítésének teljes folyamatán keresztülnyúlik, a fordítástámogatás alapvető eleme, amely a gerincét biztosítja a teljes folyamatnak. A jó munkaszervező eszköz megfelelően skálázható és bővíthető, támogatja az egyéni munkát is, de a csoportmunka előkészítési és ellenőrzési funkciói is bele vannak építve.

Csoportos fordításra általában a rövid határidők miatt van szükség. Ilyen esetben alapvető követelmény, hogy a fordításon ne lehessen észrevenni, hogy az nem egy fordító munkája. Még a jó fordítók között sem általános, hogy jól dolgoznak csoportban is, mivel a stílusuk, szóhasználatuk, a világ szegmenseiről alkotott képük különbözik. A csoportos fordítás támogatása nem merül ki a terminológiakezelésben, mint ahogyan azt a piacon kapható CAT-eszközök feltételezik. A fordítás előkészítése során rendkívül fontos a terminológia *felismerése:* annak meghatározása, hogy milyen szavakat, kifejezéseket kell terminológiának tekinteni. Ez jelenleg úgy történik, hogy egy vezető fordító vagy terminológus a fordítás előkészítése során végigolvassa az eredeti szöveget, kijelöli annak terminusait, és meghatározza a célnyelvi megfelelőit. Ez a művelet azonban időigényes, rövid szövegek esetében jól működik, de a legjobb terminológusok kapacitása sem haladja meg napi 100 oldal előkészítését. Szükség van egy olyan eszközre, amely a szöveget „átolvassa", és felismeri a szövegben található terminusokat.

A terminusok felismerése azonban nem egyszerű feladat, a közhiedelemmel ellentétben nem elegendő csak az adott szöveg szavainak gyakorisága. A terminuskeresés két módszere a statisztikai és a determinisztikus-heurisztikus módszer. A determinisztikus-heurisztikus módszerrel azokat a kifejezéseket keressük, amelyek környezetében nagy valószínűséggel terminológia szerepel, például „.... alatt azt értjük, hogy ....", „definíció: ....", „.... nevet adták neki" stb. A statisztikai módszer lényege a gyakorisági alapon történő keresés, de a kritikus gyakoriság meghatározása azért nehéz feladat, mert ez az érték szakterületenként és célközönségenként változó. Jelenleg olyan

eszközt fejlesztünk, amely minden szöveg esetében – lehetőség szerint – négy korpusszal dolgozik: egy forrásnyelvi általános, egy forrásnyelvi szaknyelvi, egy célnyelvi általános és egy célnyelvi szaknyelvi korpusszal, és ha létezik ilyen, egy kétnyelvű általános és szaknyelvi szótárral. Alapfeltevésünk, hogy a fordító számára az a terminológiai szójegyzék a legnagyobb segítség, amely olyan kifejezésekre ad egyértelmű fordítást, amilyen nem szerepel a szótárakban vagy amilyen több értelemben szerepel a szótárakban, de az adott szövegben csak egy értelemben alkalmazható. Az algoritmus alapja, hogy kiszámoljuk, hogy a potenciális terminus milyen gyakorisággal szerepel a forrásnyelvi általános korpuszban és a szakkorpuszban, kiszámítjuk ugyanezt az értéket a szótári bejegyzések lehetséges fordításai alapján a célnyelvre is, és ha az egyik fordítás esetében ez az érték kiugró, azt a kifejezést terminusnak tekintjük. A rendszer azonban csak jó korpusszal és szótárakkal működőképes, amelyek építése erőforrás-igényes munka, ezért a szakterületekre jellemző „terminusküszöbértékek" kiszámítása csak hálózati szolgáltatásként képzelhető el. A küszöbérték utána a felhasználó által finomítható. Az ideálisnál alacsonyabb küszöbérték esetén olyan kifejezéseket is terminusnak minősít az eszköz, amelyek következetes fordítására esetleg nincs feltétlen szükség, magasabb küszöbérték esetén pedig előfordulhat, hogy nem talál meg a rendszer olyan kifejezéseket, amelyek a terminológia részét kellene, hogy képezzék. A terminológiagyűjtés végső fázisában a statisztikai és a determinisztikus-heurisztikus módszerrel egymás találatai verifikálhatók. Az ilyen eszköz megkönnyíti a terminológus dolgát, hiszen viszonylag jó terminológiai konzisztencia garantálható rövid időn belül. A fordítási minőség-javító funkciója legszembetűnőbb a rendkívül hosszú szövegek nagyon rövid idő alatt, sok fordítóval történő fordítása esetén.

A munkaszervező keretrendszerbe egyéb eszközök is beépülhetnek, amilyenek például a kollokációellenőrzés, a terminológiai konzisztencia ellenőrzése, a hivatkozások eredethű fordításának ellenőrzése stb.

Irodalomjegyzék

AUSTERMÜHL, Frank (2001): Electronic Tools for Translators. Manchester: St. Jerome.

CASTELLVÍ, M. Teresa Cabré – BAGOT, Rosa Estopà – PALATRESI, Jordi Vivaldi: Automatic Term Detection: A Review of Current Systems. In: *Bourigault, Didier – Jacquemin, Christian – L'Homme, Marie-Claude (eds.): Recent Advances in Computational Terminology.* John Benjamins, Amsterdam-Philadelphia, 2001. pp. 53–88.

ESSELINK, Bert (2001): A Practical Guide to Localization, Amsterdam & Philadelphia: John Benjamins. 488 pp.

JACQUEMIN, Christian (2001): Spotting and Discovering Terms through Natural Language Processing. The MIT Press, Cambridge, MA, USA–London.

KIS, Ádám–KIS, Balázs (2003): A Prescriptive Corpus-based Technical Dictionary. Development of a multi-purpose technical dictionary. In: *Proceedings of COMPLEX 2003*, Budapest.

KLAUDY Kinga (2003): Fordítástechnikai minimum (kézirat). Budapest-Miskolc.

PRÓSZÉKY Gábor (2002): Nyelvi technológiák és gépi fordítás. *In: Emberi és gép nyelv, beszéd és hallás* (megjelenés alatt)

PRÓSZÉKY Gábor–KIS Balázs (1999): Számítógéppel – emberi nyelven. SZAK Kiadó, Bicske. 344 pp.

A New Approach To Computer-Assisted Human Translation

Balázs Kis, István Lengyel

MorphoLogic
{kis,lengyel}@morphologic.hu

The article presents methods to fight off major limitations of CAT tools currently available on the market. These limitations include language-independence and CAT tools delivered as framework products without databases.

Three types of tools or operations are concerned; only the first two are considered language tech-nology tools in the narrow sense:

(1) Translation memory: recycles previous translations of the translator or the translation community;

(2) Terminology management system: efficiently looks up and handles terminology specific to the translation domain;

(3) Workflow management tool: chops up long texts, manages translation „threads" (texts or parts of long texts where one translator and optionally one proofreader is involved), monitors and archives translations, measures efficiency, etc.

A translation memory is meant to find equivalent or similar matches to the segments of the current source text in the database of previous translations. In most cases, only similarities are found; it is crucial to have high quality hits constituting real linguistic similarity. The authors point out the need to implement the parsing of morphology and a limited syntax of the source language in CAT tools. Moreover, it would be a breakthrough in translation memories if they did not only show the exact hit in the database, the „canned translation", but also attempted to adapt the translation to the context according to the nature of similarity between the source and the stored segments. Translation memories currently available are all delivered with an empty database. However, the subject of source texts, respective vocations and sciences – and their use of language – exist independently: it would be very useful if the CAT tool had access to a general corpus of the given context, preventing the translation memory from appearing to be empty at the time of installation.

Current terminology management systems treat terminology as a rule, considering technical terms as unambiguous. This approach has many limitations. The authors' experience in translation and terminology definition shows that the major feature of terms is not disambiguity but the topic-specific formation of the context. Also, words and expressions present in everyday language are often attributed a different meaning (in another context, perhaps with a different syntax); these also have to be treated as terms. Therefore, terminology is not necessary nominal, nor unambiguous.

Also, it is very important to provide a means to spot terms: translators and translation team managers cannot be expected to know all topics they have to deal with in great detail. Recognising terminological situations – situations where a given word or collocation should be considered technical term – is not trivial and cannot be performed independently of general corpora and glossaries of the domain. The authors present the development of a terminology spotting network service.

Angol időhatározói NP-k a gépi fordításban

Ugray Gábor, Ujvárosi Gábor

MorphoLogic Kft.
{ugray,ujvarosi}@morphologic.hu

Absztrakt. Az időhatározói NP-k felismerése az elemzés során kulcsfontosságú a helyes gépi fordítás szempontjából, s mivel sajátos belső szerkezettel illetve disztribúcióval rendelkeznek, a szerzők ezek lefedésére különálló, kézzel kódolt szabályokat hoztak létre. Időhatározó-nyelvtanunk jelenleg több mint 800, példamondatokkal annotált szabályt tartalmaz. A témában rendelkezésre álló irodalom igen szűkös, különösen ami a csupasz időhatározói NP-ket illeti. A munka során szerzett tapasztalataink hatására számos általánosan határozószóként számon tartott szó besorolását megváltoztattuk, és az NP-knek és PP-knek az angol nyelvtanokban hagyományosan éles elkülönítése is megkérdőjeleződött.

1. Bevezető

A mód-, hely illetve időhatározói értelmű prepozíciós csoportok (PP-k) megkülönböztetése a szintaxis számos alkalmazásánál nem bír különleges jelentőséggel, így nem meglepő, hogy az angol nyelv egyszerű generatív nyelvészeti leírásai (Radford 1988) az alábbiakhoz hasonló szabályokból építkeznek: V' → V (NP) (PP) és V'' → V' (PP). A gépi fordítás szempontjából azonban kulcsfontosságú, hogy a szabad módosítóként álló PP-t (pl. *in June*) helyesen értelmezzük időhatározóként a – legalábbis a mi rendszerünkben – elsődlegesebb helyhatározó helyett.

A szabad módosítóként álló PP főnévi fejének lexikai tulajdonságai nyilvánvaló módon kitüntetett szerepet játszanak abból a szempontból, hogy a PP időhatározói funkciót tölt-e be a mondatban. Az *at the bus stop* szerkezet a főnévi fej lexikai tulajdonságaiból kifolyólag lesz helyhatározó, míg az *at six o'clock* a *six o'clock* NP tulajdonságai miatt működik időhatározóként.

Az alábbi időhatározói szerkezeteket megvizsgálva kitűnik, hogy a bennük szereplő NP-k igen különleges tulajdonságokkal rendelkeznek:

1. after July 4
2. at 6.30 am
3. on May 27, 1978
4. in 2003
5. before last year

Világos, hogy az 1-4. példában szereplő, dátumot ill. időt leíró ún. *clock-calendar NP*-k sajátos szerkezettel bírnak, amit a „produktív", a vonásszámot bal- illetve jobb-

oldali komplementumok és módosítók hozzávételével növelő szabályok nem képesek leírni. Ebből az alapvető megfigyelésből kiindulva a produktív NP-nyelvtan mellett elkezdtünk kézzel írt szabályokban olyan NP-ket kódolni, amelyek nagyon kevés általános, alulspecifikált elemet tartalmaznak és „kilapított" struktúrákat írnak le, azaz jobboldalukon többnyire lexikálisan megkötött terminálisokat tartalmaznak. A fenti 3. példát leíró szabály, vázlatosan:

```
(Sz1) NP = N(lex="May") + NUM + PUNCT(lex="comma") + NUM
```

Továbbmenve, az 5. példában szereplő NP-nek az angolban egyáltalán nem „szabadna" előfordulnia, hiszen megszámlálható főnév egyes számban nem állhat determináns nélkül.[1]

2. Csupasz adverbiális NP-k[2]

Az időhatározói PP-kben előforduló NP-k szokatlan szerkezete valójában csak a nehézségek kezdete. Mint azt korábban már megfigyelték (Larson 1985; McCawley 1988; Bresnan 2000), bizonyos főnévi konstrukciók prepozíció nélkül is betölthetnek határozói szerepet:

6. A strange man knocked at our door one day.
7. The country's GDP grew by 4.5 percent last year.
8. John was reading the papers all morning.
9. The president said Sunday he didn't support these plans.

Larson és McCawley saját nyelvészeti paradigmájukon belül megpróbálják leírni ezt a jelenséget. Az egyik megoldás, hogy a fejként álló főnevet olyan lexikai jellemzővel ruházzák fel, ami révén az egész NP automatikusan esetet kap; a másik, hogy ezeket a csoportokat ε-prepozíciós PP-nek tekintik. Mi egyik értelmezés mellett sem foglalunk állást. Célunk, hogy saját rendszerünkön belül a lehető legtöbb időhatározót[3] lefedjük, ezért a kézzel írt szabályok közé elkezdtünk csupasz NP-adverbiálisokat is felvenni.

---

[1] Említésre méltó, hogy a *next* ill. *last* melléknevek egyfajta korlátozott determinánsi funkcióval is rendelkeznek. Míg a *next door* és a *last year* teljesen helyesek, a **small door* és a **long year* már nem grammatikus NP-k. A helyzetet tovább bonyolítja, hogy csak bizonyos főneveket determinálhat ez a két melléknév, ld. **next house*, **last moment*.

[2] A *csupasz adverbiális* NP kifejezést az angol nyelvű szakirodalomban alkalmazott *bare adverbial NP* megfelelőjeként használjuk.

[3] A *határozó* kifejezést az angol *adverbial*-hez hasonlóan tág értelmezésben használjuk, azaz így nevezünk minden határozói értelmű szerkezetet, nem csak a határozószókat ill. azok kiterjesztéseit.

3. Kategorizálás

Az olyan kifejezések, mint a *one day, last year* és *all morning* nyilvánvalóan a főnévi csoportokra jellemző szerkezettel rendelkeznek, így természetes döntésnek tűnt, hogy akként is írjuk le őket, bár ez felveti azt a problémát, hogy hogyan tegyük lehetővé egy NP-nek, hogy időhatározói módosítóként viselkedhessen. Az ilyen NP-k disztribúciója további érveket szolgáltatott döntésünk helyessége mellett:

10. I didn't want to tell you the secret before tomorrow.
11. I lived here during last year.
12. This morning was fine.

A 10. ill. 11. mondatban a *tomorrow* és a *last year* PP-kben szerepelnek, az 12. példában pedig a *this morning* a mondat névszói alanya. Ezt a viselkedést igen nehéz lenne megmagyarázni, ha akár határozói, akár prepozíciós csoportként kategorizálnánk őket, ami a másik két lehetőség.

Még egy lépéssel továbbvive ezt a gondolatmenetet, az olyan szavakat, mint a *yesterday, today* és *tomorrow* szintén főnévként – illetve a jelenlegi leírásban NP-ként – kell kezelnünk.[4] Hagyományosan ezzel szemben feltehetőleg habozás nélkül határozószóként tüntetnénk fel őket a lexikonban.

4. A nyelvtan

Időhatározó-nyelvtanunk jelenlegi állapotában több mint 800 szabályt tartalmaz. Tesztelési célból minden szabályhoz legalább egy példamondat tartozik. Mint fent említettük, az esetek nagy részében „lapos" NP-ket írunk le,[5] azaz a csoport belső szerkezetéről nem, vagy csak minimális mértékben veszünk tudomást. A produktív szabályokkal való átfedésnek köszönhetően sok ilyen NP-nek két elemzése is születik, azonban a határozó-nyelvtan által létrehozott elemzéseket egy kitüntető jeggyel megkülönböztetjük, s megfelelő felülbírálásokkal a teljes mondatnak nem keletkezik két értelmezése.

Az időhatározói NP-k csökevényes struktúrájú, kevés általánosítást tartalmazó, lexikálisan specifikált szabályokkal történő kódolásának fő oka, hogy fordításuk nagy mértékben megjósolhatatlan. Ennek tükrében az alkalmazott „nyers erő módszere" tűnt a leggyümölcsözőbb útnak. Corley és Haywood (2000) ezen kívül rámutatnak,

---

[4] Érdekes megfigyelni, hogy bizonyos csupasz adverbiális NP-k szerepelhetnek mondat névszói alanyaként ill. prepozíciós csoportban (*Yesterday was fine, She lived elsewhere before last year*), más, nyilvánvalóan NP-szerkezetet mutató kifejezések ezt nem engedik meg (**All morning was fine*).

[5] Ez természetesen nem vonatkozik azokra az esetekre, ahol a kifejezés disztribúciója nem mutat főnévi csoportra jellemző jegyeket, mint pl. az idiomatikus *every now and then*. Az ilyeneket határozói csoportként kódoltuk.

hogy a csupasz adverbiális NP-k különleges helyet helyet élveznek az emberi nyelvtani reprezentációban is.

A nyelvtanunk egyéb területein is alkalmazott elv, hogy sok lexikális információt a terminálisok helyett magasabb szinteken kódolunk. Egy adott ige szubkategorizációit például VP-t létrehozó szabályok formájában soroljuk fel, amik az igére lexikai kikötést tartalmaznak, és jobboldalukon szerepelnek annak komplementumai. Az elemzőszabályhoz tartozó generálószabályokban a komplementumok megkapják a megfelelő magyar esetet, névutót stb. Mindezt az információt jóval egyszerűbb különálló, a komplementumokat is magukban foglaló, lexikálisan specifikált VP-s szabályok formájában leírni, mint az igéhez tartozó egyetlen lexikon-bejegyzésbe süríteni. Ugyanezt az elvet követjük, amikor egyedi, bonyolult lexikai jegyeket hordozó főnevek helyett egész NP-t tartalmazó szabályokat írunk le.

Az alábbiakban néhány vázlatos szabályon keresztül bemutatjuk az időhatározó-nyelvtanban kódolt NP-k fontosabb attribútumait. A példák csak az elemző oldali részt tartalmazzák, a generáló szabályokra helyszűke miatt nem térünk ki.[6]

```
(Sz2)  *NP=last+year:0208220945-1
       EN.NP[temp=YES, mayadv=YES, timep=YES, genadj=YES] =
       ADJ(lex="last") + N(lex="year", num=SG)
```

A (Sz2) a *last year*-t reprezentálja. A **temp** a megkülönböztető jegy, aminek **YES** értéke azt jelenti, hogy az adott NP-t az időhatározó-nyelvtanhoz tartozó szabály hozta létre. A **mayadv** azt fejezi ki, hogy csupasz adverbiális NP-ről van szó. Az ilyenek megengedik a következő szabály működését, ami már egy időhatározói szerepű szimbólumot hoz létre:

```
(Sz3)  *ADVP=NP:0206261835-1
       EN.ADVP[lexical=YES, ppreo=YES, pfinal=YES, prnp=NO, pinit=YES,
       type=TEMP, ttemp=YES] = NP(temp=YES, mayadv=YES)
```

A **genadj** szintén az NP disztribúciójáról tartalmaz információt. Nyelvtanunkban az időhatározói NP-k illetve a belőlük képződött PP-k négy helyen fordulhatnak elő: i) VP jobboldali módosítójaként, ld. (Sz3) és 13.; ii) NP jobboldali módosítója, ld. (Sz8) és 14.; iii) NP baloldali módosítója, ld. (Sz6) és 15.; valamint iv) NP birtokos determinánsa, ls. (Sz4) és 16. A **genadj YES** értékének jelentése, hogy az NP állhat állhat birtokos determinánsként.

13. He arrived last year.
14. The excursion last year was a lot of fun.
15. The six o'clock train was late again.
16. Last year's excursion was a lot of fun.

---

[6] A szabályok első sora fejléc, a nyelvtani értelemben vett információt a második sor tartalmazza, amely a nyomdai korlátok miatt itt több sorba törik. (Sz2) a megszokott írásmóddal az NP → ADJ N alakú szabálynak felel meg. A szögletes zárójelek közötti értékadásokban a létrejövő szimbólum jegyeinek alapértelmezés szerinti értékeit bíráljuk felül; a jobboldali szimbólumok után álló kerek zárójelek között pedig azok jegyeire teszünk kikötéseket.

Az egyik szabály, ami lehetővé teszi, hogy időhatározói NP-k más NP-k birtokos determinánsaként szerepelhessenek az (Sz4):

```
(Sz4)  *NPX=NP(adj)+S+NM:0304170339-10
       EN.NPX[...] = NP(genadj=YES, temp=YES) + GENS + NM(...)
```

Fontos megkülönböztetni ezt az esetet a *the man's wife*-hoz hasonló „valódi" birtokos szerkezetektől. Míg az utóbbiak magyarban is birtokos szerkezetre fordulnak, az előbbiekből jelzői szerkezetet képzünk (*a tavalyi kirándulás).*

A mayadj jegy azt fejezi ki, hogy az NP egy főnév baloldali módosítójaként állhat, mint a 15. példában. Az itt szerepet játszó szabályok:

```
(Sz5)  *NP=NUMX+oclock:0206081108-11
       EN.NP[temp=YES, mayadj=YES, timep=YES, timet=HR, ofradj=YES,
       timeprep="at"] = NUMX(numtype=CARD) + N(lex="o'clock")

(Sz6)  *ADJY=NP:0208081727-1
       EN.ADJY[type=TIME] = NP(temp=YES, mayadj=YES)

(Sz7)  *NN=ADJP+NX:0205291926-1
       EN.NN[...] = ADJP + NX(...)
```

A csupasz adverbiális NP-k jobboldali módosítóként való elemzését lehetővé tevő szabály:

```
(Sz8)  *RADJ=timeNP:0302110214-2
       EN.RADJ[radjuid="adv", hupos=LEFT] = NP(temp=YES, mayadv=YES)
```

Azok az időhatározói NP-k, amelyek prepozícióval együtt fordulnak elő, többnyire három csoport valamelyikébe esnek aszerint, hogy a „mikor?" kérdésre adott válaszhoz az *in, or* ill. *at* közül melyiket választják. Ennek illusztrálására szolgáljon az alábbi három példa, valamint a hozzájuk tartozó szabályok:

17. The accident happened on August 23.
18. There was no rain in the spring.
19. John's train arrived at 6 am.

```
(Sz9)  *NP=MONTH+DAY:0207111442-2
       EN.NP[temp=YES, mayadv=YES, mayadj=YES, timet=DATE, timep=YES,
       timeprep="on", num=SG, pers=P3, ofradj=YES] = NX(month=YES,
       num=SG) + NUMX(numtype=CARD, middle=YES)

(Sz10) *NP=the+season2:0208281300-1
       EN.NP[temp=YES, timep=YES, timeprep="in"] = DET(dettype=DEF) +
       NX(season=YES, lex!="winter",lex!="summer")

(Sz11) *NP=NUMX+am:0206061754-2
       EN.NP[temp=YES, mayadj=YES,timet=HR, timep=YES, timeprep="at"] =
       NUMX + ADV(lex="am")

(Sz12) *ADVP=PREP+NP(timep):0207021632-1
       EN.ADVP[lexical=YES, pfinal=YES, pinit=YES, prnp=NO, type=TEMP,
       ttemp=YES](PREP.lex=NP.timeprep) = PREP + NP(temp=YES)
```

A **timeprep** jegyben tároljuk az NP által választott prepozíciót. Amennyiben üresen marad – s ez az alapértelmezett érték –, a (Sz12) nem sülhet el, mert a baloldalra írt feltétel, hogy a prepozíció szótári alakja egyezzen meg az NP **timeprep** jegyében tárolt értékkel, nem teljesül. Létezik egy hasonló szabály, ami révén a prepozíciót igénylő NP-k előfordulhatnak főnév jobboldali módosítójaként (ld. 20. példa), bár tapasztalatunk szerint nem minden VP-módosítóként megfigyelhető PP működhet főnév jobboldali módosítójaként is.

20. The meeting at 6 am was quite a thrill.

Bizonyos prepozíciók (például a *before, for, during* stb.) előfordulása az NP más lexikai tulajdonságától függ.

```
(Sz13) *ADVP=after:0206260909-2
       EN.ADVP[lexical=YES, pfinal=YES, pinit=YES, prnp=NO, type=TEMP,
       ttemp=YES] = PREP(lex="after") + NP(temp=YES,  timep=YES)

(Sz14) *ADVP=for+NX(dura):0303071309
       EN.ADVP[lexical=YES, prnp=NO, pinit=YES, pfinal=YES, ppreo=YES,
       type=TEMP, ttemp=YES] = PREP(lex="for") + NP(temp=YES, dura=YES)
```

A **timep** jegy durván azt fejezi ki, hogy a szóban forgó NP egy „időpontra" vonatkozik, míg a **dura** jelentése, hogy „időtartamot" ír le.

A *for* és *in* prepozíciók különleges figyelmet igényelnek, mivel kétféle szemantikai értelmezésük is létezik. (Hitzeman 1996) amellett érvel, hogy ezek függetlenek az igeidőtől, a mi céljainkra azonban „elég jó" heurisztikának megfelel, hogy megkülönböztetésül az igeidőre támaszkodjunk. Az elemzési oldalon pedig a két értelmezés között szintaktikai szempontból egyáltalán nem mutatkozik különbség.

5. Hivatkozások

1. Bresnan, J.: Lexical-Functional Syntax. Blakwell Publishers, Oxford (2000)
2. Chomsky, N.: Remarks on Nominalization. In: R. Jacobs and P. Rosenbaum (szerk.): Readings in English Transformational Grammar. Blaisdell Publishing, Waltham, Massachusetts (1970)
3. Corley, M. & Haywood, S.: Parsing modifiers: The case of bare-NP adverbs. In: Proceedings of the 21st annual meeting of the Cognitive Science Society, 126-131. Vancouver (2000)
4. Hitzeman, J.: Semantic partition and the ambiguity of sentences containing temporal adverbials. Online: http://www.hcrc.ed.ac.uk/publications/rp-77.ps.gz (1996)
5. Komlósy, A.: A lexikai-funkcionális grammatika mondattanának alapfogalmai: Segédkönyvek a nyelvészet tanulmányozásához VII.; Kiss Gábor (szerk.)/ Nem transzformációs nyelvtanok I. Tinta Könyvkiadó, Budapest (2001)
6. Larson, R: Bare-NP Adverbs. In: Linguistic Inquiry 16, 595-621. (1985)
7. McCawley, J D: Adverbial NP's: bare or clad in see-through garb? In: Language 64, 583-90. (1988)
8. Radford, A: Transformational Syntax. Cambridge University Press, Cambridge (1988)

English Time Adverbial NP's in Machine Translation

Ugray Gábor, Ujvárosi Gábor

MorphoLogic, Ltd.
{ugray,ujvarosi}@morphologic.hu

Keywords: machine translation, time adverbials, adverbial NP's

Recognizing time adverbial NP's in the parse side is crucial for correct MT, therefore this issue has received special attention during the development of our English-Hungarian machine translation system. This paper describes our solution and presents some theoretical conclusions we have arrived at during the formalization and lexical coding of time adverbial structures.

Most models of natural language, especially generative syntax, tend to stop at prepositional and adverbial phrases for modifiers of the nominal or verbal phrase. The translation of prepositional phrases that serve as time adverbials is largely unpredictable, therefore they need to be individually coded. In the course of this work it has become obvious that the key issue is the description of the contained NP's.

In this paper we present some of the structural and distributional properties of time adverbial NP's. We point out structural peculiarities (e.g., *last night* is grammatical, while **last house* is not) and describe the set of pertaining lexical features used in our grammar.

Our solution is novel in the sense that we encode lexical properties as features of the NP itself instead of its head noun, which means that we store whole NP's with little internal structure or generalization. From the point of view of parsing, this decision was motivated by the irregular structure and distribution of the NP's in question, and from the point of view of MT, by the unpredictability of the translation.

The paper discusses the phenomenon of bare adverbial NP's, that is, noun phrases that can serve as adverbials of time even in the absence of a preposition. This is an area with scarce literature, and our experiences have lead us to rethink the traditionally sharp distinction between PP's and NP's in English syntax.

Rövid előadások

Gépi beszédfelismerők betanítása – Mennyi kézi szegmentálásra van szükségünk?

Mihajlik Péter, Tatai Péter, Gordos Géza
BME,Távközlési és Médiainformatikai Tanszék
mihajlik@tmit.bme.hu

Kulcsszavak: automatikus beszédfelismerés, beszélőfüggetlen telefonos felismerő tanítás, beszédadatbázis, kézi – gépi szegmentálás, fonetikus átírás

A mai beszélőfüggetlen beszédfelismerők betanításához nagy mennyiségű beszédadatra van szükség. Mivel a felismerés elemi egységei tipikusan a beszédhangok, feladatunk ezen szegmentumok és pozíciójuk meghatározása a tanítóanyagban, hogy a megfelelő hangrészletekkel a megfelelő beszédhang-modellek betaníthatók legyenek. A tanító-adatbázis hangokra szegmentálása történhet kézi vagy automatikus módszerekkel, illetve ezek kombinációival. Míg a világtrend szerinti "főáramlatban" szinte csak (implicit) gépi szegmentálást használnak, néha nagy mennyiségű kézi szegmentálást tartalmazó beszédadatbázisok is kifejlesztésre kerülnek, mint pl. a nemrégiben elkészült MTBA (Magyarnyelvű Telefon-Beszéd Adatbázis)[1].

Fontosnak éreztük eldönteni, hogy a felismerés pontosságát javítandó, melyik utat válasszuk; ezért az említett adatbázison kutatásokba kezdtünk. Először mind az 500 beszélő kézi szegmentálását felhasználva tanítottuk a rendszert, környezetfüggő beszédhangmodelleket alkalmazva. Így 1000-es szótárméretű, izoláltszavas felismerési feladat esetén, független teszt adatbázison 6.85%-ra tudtuk szorítani a felismerési hibát. Ezután megpróbáltuk mindössze 10 beszélő kézi szegmentálásának felhasználásával betanítani a felismerőt. Összetett tanítási módszerünk a továbbiakban már csak az (500 beszélős) adatbázis annotált szövegeit és hullámformákat igényelte. Kiejtési változatokat is tartalmazó fonetikus átiratokat fonetikai szabályok alapján automatikusan állítottuk elő[2]; ezek segítségével a "kényszerített felismerés" módszerét alkalmazva automatizáltuk a lehallgatást és a szegmentálást is. A végeredményül kapott felismerési hiba mindössze 7.02% lett.

Összegzésként megállapíthatjuk, hogy mélyebb nyelvi információkat is felhasználó tanítási módszerünket alkalmazva gyakorlatilag ugyanannyi a felismerési hiba, mint a kizárólag kézi szegmentáláson alapuló esetben. Ugyanakkor a szükséges kézi munka mennyiségét az eredeti *50-ed részére* csökkentettük.

[1] http://alpha.ttt.bme.hu/speech/MTBAhun.htm

[2] Mihajlik, P., Révész, T. and Tatai, P., Phonetic transcription in automatic speech recognition, Acta Linguistica Hungarica, Vol. 49, pp. 407–425, 2002

Speech Recognizer Training – How Much Manual Segmentations Do We Need?

Péter Mihajlik, Péter Tatai, Géza Gordos
Department of Telecommunications and Media Informatics, BUTE
mihajlik@tmit.bme.hu

Keywords: Automatic speech recognition, speaker-independent telephone speech recognition, training, speech database, transcriptions, segmentations

Today's speaker-independent automatic speech recognizers require hundreds of hours of training speech data. The basic units of recognition are typically the speech sounds; therefore these units and their positions have to be identified in the database. This process is called phonetic segmentation and can be performed either manually or automatically. While the worldwide approach is to perform the (implicit) phonetic segmentation entirely automatically, sometimes a big amount of manual segmentations are made, see e.g., the recently collected HTSD - Hungarian Telephony Speech Database.[1]

We conducted some experiments on one of the mentioned database (HTSD) in order to determine the optimal approach to database development towards increasing the efficiency of speech recognizers. First we trained the ASR system based on the manual phonetic segmentations of the 500 speakers. Context dependent (CD) phone models were used. The recognizer was tested on independent test data; the task was isolated word recognition with a vocabulary size of 1000. The best recognition error rate we could achieve was 6.85%. Then we trained the recognizer based on the segmentations of only 10 speakers by a sophisticated training method developed at our laboratory. The process needed only the annotations and the waveforms of the 500 speakers additionally. Phonetic transcriptions containing pronunciation alternatives were generated automatically based on phonological rules[2]. These special transcriptions were applied at the forced alignment phase where the actual phonetic realizations and the phone boundaries were determined automatically. So, effectively, the human listener was replaced by the computer. Finally, we obtained an error rate of 7.02% on the reference recognition task.

We can conclude that using our training method that utilizes deeper phonological knowledge the recognition error rate is practically the same as in the case of fully manual segmentation. At the same time the required amount of the highly expensive manual segmentations was reduced by *50 times*.

---

[1] http://alpha.ttt.bme.hu/speech/MTBA.htm

[2] Mihajlik, P., Révész, T. and Tatai, P., Phonetic transcription in automatic speech recognition, Acta Linguistica Hungarica, Vol. 49, pp. 407–425, 2002

A magyar nyelv ejtésvariáció vizsgálata gépi beszédfelismerés segítésére

Szaszák György[1] – Vicsi Klára[1]

[1] Budapesti Műszaki és Gazdaságtudományi Egyetem, Távközlési és Médiainformatikai Tanszék, Beszédakusztikai Kutatólaboratórium. 1111 Budapest, Sztoczek u. 2.
szaszak@tmit.bme.hu , vicsi@tmit.bme.hu
http://alpha.ttt.bme.hu/speech/

Kulcsszavak. Beszédfelismerés, ejtésvariáció-modellezés, nyelvi modell.

Előadásunkban a számítógépes beszédfelismerés fonológiai szintjén használható ejtésvariáció modellezésről szeretnénk áttekintő képet nyújtani. Az emberi beszédet – különösen annak spontán, társalgásbeli formáját – a kiejtés szempontjából nagyfokú variáltság jellemzi, azaz ugyanazon szöveg(rész) többféle, de egyaránt helyes kiejtéssel is előfordulhat. A beszédfelismerők tévesztéseiben ennek a jelenségnek is nagy szerepe van, az ejtésvariáció modellezés célja tehát a felismerés hatékonyságának növelése.

Magyar nyelvre ejtésvariáció vizsgálatokat a Magyar Telefonbeszéd Adatbázis anyagán végeztünk, adatalapú, azaz statisztikai megközelítést használva, de célunk ennek alapján kiejtési szabálybázis megkonstruálása is. A vizsgálati metodika röviden a következő: elegendően nagyszámú beszélő által bemondott és speciálisan megszerkesztett szöveg hanganyaga van rögzítve, amely reprezentálja a magyar köznyelv bemondási variációit. A vizsgálat során a bemondott szöveg kanonikus ejtésének megfelelő fonetikus átírást összehasonlítjuk a szegmentálás során szakértők által elkészített audio-vizuális fonetikai átírattal. Az eredményeket mátrixos formában kiértékelve kaphatjuk meg. Kanonikus kiejtés alatt egy referneciaként elfogadott (célszerűen a leggyakoribb) kiejtés értendő. A vizsgálatot beszédhangokra, majd hangkapcsolatokra végeztük el.

Az ejtésvariáció-vizsgálatok során nyert eredmények felhasználási lehetőségeiről az alábbiakat mondhatjuk el. Amellet, hogy az ejtés variáltságának mint jelenségnek beszédképzésbeli, illetve nyelvi, sőt információelméleti összefüggéseit is megismerjük, az eredmények közvetlenül két fő vonalon használhatóak fel. Az első lehetőség a felvételek szegmentálása és címkézése során használt fonetikai átírási szabályrendszer, illetve szimbólumkészlet kontrollálása, a második lehetőség a szerzett ismeretek beszédfelismerés során történő hasznosítása, a felismerés hatékonyságának növelése az ejtésvariációk modellezése révén. Mindennek jelentősége különösen a nyelvi modellezés során van, hiszen ehhez pontosan ismernünk kell az egyes kiejtésalakok gyakoriságát vagy előfordulásuk szabályszerűségeit.

Pronunciation variation modeling of Hungarian language for CSR

György Szaszák[1] – Klára Vicsi[1]

[1] Budapest University of Technology and Economics, Department of Telecommunication and Mediainformatics, Laboratory of Speech Acoustics. 1111 Budapest, Sztoczek u. 2.
szaszak@tmit.bme.hu , vicsi@tmit.bme.hu
http://alpha.ttt.bme.hu/speech/

Keywords. speech recognition, pronunciation variation, language modeling

In our presentation we provide an overview about pronunciation variation modeling used in Continuous Speech Recogniser (CSR) systems at the phonological level. Human speech is caracterized by a high degree of variability with regard to pronunciation forms, mainly in case of spontaneous, everyday speech. This means that the same text can be uttered in more then one correct manner. Pronunciation variability in speech is a not negligible source of misrecognitions in CSR systems, attempts to model pronunciation variation envisage the reduction of the error rates and to get more insight into linguistic mechanisms which control pronunciation variation during speech production.

By examining pronunciation variation in Hungarian language, we used a data-driven method, but we are planning to construct a knowledge base from our current results. The examination procedure was as follows: we examine whether the actual pronunciation of a phone corresponds to its canonical pronunciation, that is regarded as a reference one, obtained by rule based automatic phonotypical transcription of the corpus. The actual pronunciation was obtained by audio-visual transcription of the utterances. By comparing these two streams we get statistics about pronunciation variants of each sound, presented in matrice form. This examination process strated with separate phones, then it was extended to biphones or triphones too.

The importance and further benefits of pronunciation variation examinations are: we get better insight into the governing mechanisms of pronunciation variation so we can investigate the relationship among linguistic or grammatical aspects of speech and pronunciation variation. More practical advantages are the feedback to automatic rule based phonotypical transcription, which means the control and the correction of the phone set or the rules applied. Finally the integration of pronunciation variation models into CSR systems could improve recognition performance by ensuring a more robust and flexible background for speech recognisers mainly on the language modeling level which needs exact knowledge about the different pronunciation forms either in the form of pure statistics or as a knowledge base.

LAS Verticum: Egy szó feletti tartalomelemző szoftver

László János[1] és Ehmann Bea[2]

[1] MTA Pszichológiai Kutatóintézet
Budapest, Victor Hugó u. 18-22. H-1132 Magyarország
laszlo@mtapi.hu

[2 1] MTA Pszichológiai Kutatóintézet
Budapest, Victor Hugó u. 18-22. H-1132 Magyarország
ehmannb@mtapi.hu

Kulcsszavak. LAS Verticum, LINTAG, tartalomelemzés (szó feletti), narratív pszichológiai tartalomelemzés

A LAS vertikumot három összetevő alkotja: a LINTAG nevű szó feletti kódolóprogram, az ATLAS.TI fogalmi hálózatépítő program és az SPSS statisztikai programcsomag. Az előadásban a LINTAG és az Atlas.ti összekapcsolódásának logikáját mutatjuk be.

A LINTAG program a Morphologic Kft. által korábban kifejlesztett HUMORESK morfológiai elemzőprogram elvén alapul, amely a természetes nyelv szókészletét morfémákra bontva tartalmazza. A szoftvert elvileg éppen ez a tulajdonság teszi alkalmassá magyar nyelvű szövegek szószintű tartalomelemzésére. A LINTAG program másik fő alkotóeleme a narratív pszichológiai tartalmakkal feltöltött nyelvtani és lexikon fájlok sorozata. A szoftvernek ezt a részét az MTA Pszichológiai Kutatóintézete, a Pécsi Tudományegyetem és a Gödöllői Szent István Egyetem közös Narratív Pszichológiai Munkacsoportja fejlesztette ki a Morphologic Kft-vel együttműködve.

Ha a program csupán szógyakorisági elemző volna, akkor is újdonságot jelentene a magyar nyelvű szövegek számítógépes tartalomelemzésének terén: képes lenne arra, hogy pszichológiai relevanciával rendelkező, validált szókategóriák tagjainak előfordulási gyakoriságát kijelezze. A LINTAG szupralexikális mivolta két területen jelentkezik. Az egyik, hogy a vizsgálni kívánt szókategóriák nem csupán egyetlen szóból állhatnak, hanem több szóból álló kifejezéseket is tartalmazhatnak. A másik – és ez a lényegesebb –, hogy a program nem csupán itemenként egy vagy több tagból álló szólisták alapján, hanem szó feletti mintázatkereső algoritmusok révén is képes a szövegelemzésre.

A LINTAG bemenetek lehetnek szólisták és szó feletti mintázatkereső algoritmusok. A kimenetek ennek megfelelően szólista modulok és algoritmus modulok, amelyek úgy alkotják az Atlas.ti bemenetét, hogy néhány gombnyomás révén a szövegfájlok lekódolt formában jelennek meg az Atlas.ti felhasználói felületén. Az Atlas.ti kimenetei az SPSS potenciális bemenetei.

LAS Verticum: A Supralexical Content Analyzing Software

Janos Laszlo[1] and Bea Ehmann[2]

[1]Institute for Psychological Research of the Hungarian Academy of Sciences
H-1132 Victor Hugo str. 18-22, Budapest, Hungary
laszlo@mtapi.hu

[2] Institute for Psychological Research of the Hungarian Academy of Sciences
H-1132 Victor Hugo str. 18-22, Budapest, Hungary
ehmannb@mtapi.hu

Keywords. LAS Verticum, LINTAG, content analysis, narrative psychological content analysis

The LAS Verticum is composed of three different softwares: the LINTAG supralexical content analyzing program, and two commercial ones, the ATLAS.TI conceptual network builder program and the SPSS statistical program pack.

LINTAG is based on the principle of the HUMORESK Morphological Analyzer developed by the Morphologic Ltd. The HUMORESK Program is based on a new Hungarian grammar that includes the natural vocabulary as broken down according to morphemes. This particular feature makes the software capable of word level content analysis. The other main component of LINTAG is a series of grammar and lexicon files filled up with narrative psychological contents. This part of the software is a development of the Narrative Psychology Team formed by the Institute for Psychological Research of the HAS, the University of Pecs and The Szent Istvan University of Godollo, in cooperation with the Morphologic Ltd. All institutions are in Hungary.

LINTAG would be a novelty in the content analysis of Hungarian texts even if it were only a traditional word frequency analyzer, because it is able to indicate the frequency of validated word categories of psychological relevance.

The supralexical nature of LINTAG is manifested in two main fields. One is that its word categories are composed not only of single words, but may contain composite phrases as well. The other, more important, aspect is that the text analyzing process is based not on word categories only, but on supralexical pattern searching algorhytms.

LINTAG inputs can be word lists and supralexical pattern searching algorhytms. LINTAG outputs are word list based modules and algorhytm based modules, which then serve as inputs to Atlas.ti as automatic codes. Outputs of Atlas.ti may form inputs to SPSS.

A LAS Verticum narratív pszichológiai tartalomelemző rendszer időmodulja

Ehmann Bea[1]

[1] MTA Pszichológiai Kutatóintézet
Budapest, Victor Hugó u. 18-22. H-1132 Magyarország
ehmannb@mtapi.hu

Kulcsszavak. LAS Verticum, tartalomelemzés, narratív pszichológiai tartalomelemzés, időélmény

Az idő narratív pszichológiai vizsgálatának két alapelve, hogy a szubjektív időélmény mintázatai pszichológiai relevanciával rendelkeznek, és laikus beszélők személyes dokumentumaiból (interjúk, naplók, önéletírások, stb.) tartalomelemzés révén azonosíthatók. A pszichikus időélmény a számítógépes szövegelemzés révén három aspektusból vizsgálható. Ezek a szógyakorisági elemzés, a kontextuselemzés (KWIC = Keyword in Context) és a konceptuális vagy tematikus elemzés.

Az automatizált szógyakorisági elemzés két vezető szoftvere a LIWC (Linquistic Inquiry Word Count, Francis és Pennebaker, 1993) és a RID (Regressive Imagery Dictionary, Martindale, 1995). Mindkét program rendelkezik időmodullal, de ezek csupán az idő három klasszikus aspektusát, a múlt-jelen-jövő idejű felosztást kezelik.

A LAS Verticum időmodulja két területen jelent újdonságot. Szemléletileg azért, mert kitágítja a szubjektív időélmény vizsgálatának kereteit, strukturálisan pedig azért, mert a szoftver keresőfunkciója nem egyszerű szólistákon, hanem morfológiai mintázatokon alapul.

A modul kategóriái az idő következő minőségeit képesek azonosítani: a lineáris (naptári) időtengelyhez képest egy ponton lehorgonyzott időutalásokat, a kezdet, a tartam és a befejezettség aspektusait, valamint az olyan szubjektív idői tematizációkat, melyek szerint valamely esemény soha nem történt meg, örökké történik, ismételten történik, hosszú időn át történik vagy bizonytalan időpontban történik. További azonosítható minőségek az időélmény lelassulása, megállása és felgyorsulása, valamint az idő léptéke a másodperctől az évszázadig.

A LAS Verticum időmodulja automatikusan a fenti kategóriáknak megfelelő kódokat helyez el a vizsgált szövegekben. A kapott találatok révén megvalósul az automatizált szógyakorisági elemzés, és vizsgálható a találatok szövegkörnyezete (KWIC), az időmodul által nyert együttjárási mintázatok pedig kiindulópontként szolgálnak a konceptuális elemzéshez, azaz a laikus beszélők által mondott történetek narratív kronológiájának szerkezeti vizsgálatához.

The Time Module of the LAS Verticum Content Analysis System

Bea Ehmann[1]

[1] Institute for Psychological Research of the Hungarian Academy of Sciences
H-1132 Victor Hugo str. 18-22, Budapest, Hungary
ehmannb@mtapi.hu

Keywords. LAS Verticum, content analysis, narrative psychological content analysis, time experience

Two basic principles of the narrative psychological analysis of time are that the patterns of subjective time experience have psychological relevance and can be identified by content analysis in personal documents (interviews, diaries, autobiographical recalls, etc.) of lay writers. Psychical time experience can be studied in three aspects, namely word frequency analysis, KWIC (Keyword-In-Context) analysis and conceptual or thematic analysis.

The two leading softwares of automatized word frequency analysis are the LIWC (Linquistic Inquiry Word Count, Francis és Pennebaker, 1993) and the RID (Regressive Imagery Dictionary, Martindale, 1995). Both programs have a time module, but they handle only the three classic aspects, the past, the present and the future.

The time module of the LAS Verticum is a novelty in two fields. Theoretically, it extends the scope of the investigation of psychical time experience, and structurally, the search function of the software is based not on simple words lists, but on morphological patterns.

The categories of the Module are able to identify the following qualities of time: time references fixed to a particular point of the linear (calendar) time axis, and the aspects of start, duration and endpoint. Another subset is a subjective quality of time referring to that a particular event has never happened, happens all the time, happens repeatedly or happens at an uncertain point of time. Further qualities are the slowing, stopping or accelerating of time experience, and the scale of time from the second to the century.

The Time Module of the LAS Verticum assigns codes to the above time categories automatically in the texts. Within the System, this word frequency analysis allows KWIC analysis, and the correlating patterns form a starting point to conceptual analyses, namely to structural investigations of the narrative chronology of stories produced by lay writers.

A kapcsolati viszonyok téri szerveződésének vizsgálata

Pohárnok Melinda

PTE BTK Pszichológiai Intézet, Pécs, Ifjúság útja 6. 7624
pomel@freemail.hu

Kulcsszavak: narratív pszichológiai tartalomelemzés - interaktív tér – kapcsolati viszonyok – „közelítés-távolítás"-modul

A narratív pszichológiai tartalomelemzés módszerével megragadhatóak az elbeszélések olyan inherens, strukturális jellegzetességei, amelyek pszichológiai jelentéstartalommal bírnak. Abból a feltevésből indulunk ki, hogy az elbeszélői perspektíva és az idő dimenziója mellett ilyen jellegzetességnek tekinthető az elbeszélések szereplői közti téri viszonyok változása, mintázata. Feltételezzük, hogy létezik egy olyan interperszonális vagy interaktív tér, amely mindig az én és a másik viszonya alapján szerveződik: a tér két végpontját az én és a másik adja meg, és egymás viszonyában való mozgásuk a kapcsolat alapvető sajátságának tekinthető. Az én és a másik viszonya leírható egyrészt konkrét, fizikai térben való mozgásokkal: a felé (vele) – tőle (nélküle) dimenzióban. Másrészt az interaktív tér interszubjektív aspektusában: megosztozás (megértés) – megosztozás hiánya (nem-megértés) állapotaiban. A tárgyakapcsolati elméletek (Mahler, 1975) és a szelf-fejlődés elméletei (pl. Stern, 2002) szerint ez a dimenzió jelentős szerepet tölt be a korai kapcsolati- és szelf-szerveződésben. Az élettörténeti elbeszélésekben így alapvetővé válik az én és a jelentős másik/mások téri-kapcsolati viszonyainak elmezése. Ennek érdekében folyik a - Morphologic Kft. Által kifejlesztett – LintagTi szoftveren belül egy „Közelítés-Távolítás" modul kidolgozása a kutatócsoportban. A modul adott igei kategóriák – ún."viszony igék" – és adott főnévi kategóriák – jelentős másokra utaló főnevek – együttes kezelését teszi lehetővé. Így Közelítés – Távolítás igei csoportok, mint szövegkódok azonosítására alkalmas. A corpusban végül az AtlasTi szoftver segítségével nyílik mód a Közelítés-Távolítás kódok használatára és mintázataiknak vizsgálatára. A mintázatok a kapcsolatszerveződés, kapcsolatszabályozás pszichológiai változójának felelnek meg. A prezentációban néhány példán keresztül mutatjuk be a modul működését.

Analysis of spatial organization of interpersonal relations

Melinda Pohárnok

University of Pecs, Institute of Psychology Pécs, Ifjúság útja 6. 7624
pomel@freemail.hu

Keywords: narrative psychological contentanalysis – interactive space – interpersonal relations - „Approachment-Avoidance" module

The method of narrative psychological contentanalysis affords the opportunity to capture such inherent structural characteristics of narrative which have psychological meaning. It is presumed that beside the narrator's perspective and the temporal dimension of the narrative, the patterns and changes of spatial relations of characters could be also considered as significant narrative feature. The existence of an interpersonal or interactive space which is organised by the relation of the self to other/s is assumed. The extremities of this space would be the self and the other, and their movements in relation to each other could be regarded as the fundamental characteristic of their interpersonal relationship. On the one hand the relationship of self and other can be described as movements in the concrete, physical space: in a „toward him (with him)" – „from him (without him)" dimension. On the other hand it can be depicted as the intersubjective aspect of interactive space: as states of sharing (understanding) – lack of sharing (lack of understanding). According to psychoanalytical object-relational theories (Mahler, 1975) and theories of self-development (Stern, 2002) this dimension plays an important role in the early interactive- and self-organization. As a consequence the analysis of spatial organization of interpersonal relations has become essential in the case of biographical narratives. In the research group we are working on an „Approachment-Avoidance" module applied by the LintagTi software – developed by Morphologic Ltd. The module deals with the co-occurence of given verbal categories – so called „relation-verbs" – and given nouns – common nouns referring to significant others. So it can be used for identifying Approach-Avoidance verbal groups as text-codes. The resultant codes are applied to the corpus by the AtlasTi software. In our presentation the operation of the Approach-Avoidance module will be illustrated.

A narratív perspektíva automatikus kódolása élettörténeti narratívumokban

Pólya Tibor[1]

[1] MTA Pszichológiai Kutatóintézet
1132 Budapest, Victor H. u. 18-22.
polya@mtapi.hu

Kulcsszavak. Narratív perspektíva, deiktikus centrum

Az előadás az élettörténet narratív perspektívájának azonosítására kidolgozott nyelvi elemző modult mutatja be. A narratív perspektíva fogalma arra utal, hogy a narratívum tartalmát adó narratív elemek (események, szereplők, körülmények) mindig egy konkrét nézőpontból kerülnek bemutatásra, amely a narratívum deiktikus centrumával azonos. Az élettörténeti narratív perspektíva lehetséges formáinak meghatározásában a temporális lokalizációra építek. Azt feltételezve, hogy a deiktikus centrum és a narratív elemek temporális lokalizációja két értéke vehet fel (elbeszélt események versus elbeszélési események) a narratív perspektíva három variációja azonosítható.
Visszatekintő narratív perspektíva: a deiktikus centrum az elbeszélési eseményekhez, a narratív elemek az elbeszélt eseményekhez lokalizáltak időben.
Átélő narratív perspektíva: a deiktikus centrum és a narratív elemek is az elbeszélt eseményekhez lokalizáltak időben.
Újraátélő narratív perspektíva: a deiktikus centrum és a narratív elemek is az elbeszélési eseményekhez lokalizáltak időben.
A nyelvi elemző modul a Morphology Kft. által kifejlesztett nyelvi elemző (Lintag) kimenetét dolgozza fel. Az algoritmus két fő összetevője a deiktikus markerek és a narratív perspektívára specifikus szócsoportok elemzése. A deiktikus markerek elemzése a deixis három kategóriájára terjed ki: idő (igeidő és időbeli határozószavak), hely (helyhatározószavak és mutató névmások) és személy (személyes névmások) deixis. A deiktikus markerek elemzését kiegészíti a narratív perspektíva specifikus szócsoportok vizsgálata: visszatekintőnél a dátumra utaló kifejezések, átélőnél az indulatszavak, és újraátélőnél a szubjektív modalitás kifejezései.

Automatic coding of the narrative perspective in life story narratives

Tibor Pólya[1]

[1] Institute for Psychology of the HAS
1132 Budapest, Victor H. u. 18-22. Hungary
polya@mtapi.hu

Keywords: narrative perspective, deictic center

This presentation is about the algorithm for automatic coding of the various forms of narrative perspective in life story narratives.
The term of narrative perspective refers to the deictic center of a narrative from where the narrator presents the narrative elements (events, characters, and circumstances). Definition of the various forms of a narrative perspective can be based on the temporal location. Assuming that both deictic center and narrative elements are located either to the reported event or to the reporting event, three forms can be distinguished.
Retrospective narrative perspective: the deictic center is located to the reporting event, and the narrative elements are located to the reported event. Experiencing narrative perspective: both the deictic center and the narrative elements are located to the reported event.
Re-experiencing narrative perspective: both the deictic center and the narrative elements are located to the reporting event.
The automatic coding of a narrative perspective makes use of the output of a language parsing software (Lintag) developed by Morphology Ltd. Two main components of the algorithm are the analysis of some deictic markers and specific terms. The analysis of deictic markers includes three categories of deictic markers: time (tense and adverbs of time), place (adverbs of place and demonstrative pronouns), and person (personal pronouns) deixis. The specific terms are the followings: words explicitly referring to dates, interjections, and various expressions of a subjective modality, which are related to the retrospective, experiencing, and re-experiencing narrative perspectives, respectively.

Univerzális konfigurációs nyelv és mag-architektúra párbeszédes rendszerekhez

Kovásznai Gergely

Számítógéptudományi Tanszék, Informatikai Intézet, Debreceni Egyetem,
kovasz@inf.unideb.hu

Kulcsszavak párbeszéd menedzsment, párbeszédes ügynökprogram, multimodális interakció, XML

Napjaink tendenciája, hogy az ember-gép kommunikáció a természetes emberi kommunikációs formák irányába tolódik el, azaz az ember-ember kommunikációt veszi alapul és próbálja utánozni multimodális számítógépes környezetben. Mivel emberek között a beszélt nyelv a legfontosabb információ-átvivő közeg, a multimodális rendszerek magját is az úgynevezett párbeszédes rendszerek képezik. Ezen rendszerek szöveges inputot fogadnak, s ezen inputnak és saját belső állapotuknak megfelelő szöveges outputot bocsátanak ki. Napjaink párbeszédes rendszerei közös ismérve erősen behatárolt használhatóságuk. Valamennyien nyelv- és alkalmazásspecifikusak, s az utóbbiból eredően csak meghatározott modulokkal társíthatók egy konkrét multimodális alkalmazásban. Kutatásaink célja egy olyan párbeszédes rendszermag-architektúra és ehhez egy olyan konfigurációs nyelv megalkotása volt, melyek univerzálisak abban az értelemben, hogy nyelv- és alkalmazásfüggetlenek, valamint tetszés szerinti modulokkal bővíthetők. Továbbá szem előtt tartottuk a következőket:

- hatékony nyelvi elemzés, nyelvi tartalom generálás és szemantikus adatábrázolás, illetve ezek tetszés szerinti bővíthetősége;
- hatékony alacsony szintű procedurális absztrakció;
- könnyű használhatóság tapasztalatlan felhasználók számára is, azaz hatékony heurisztikus absztrakció.

Az általunk kigondolt konfigurációs nyelvet "Conversational Agent Markup Language"-nek (CAML) neveztük el, valamint a hozzá társított architektúrát CAML magnak. A CAML egy XML-alapú nyelv, mely ún. kategóriák definiálására szolgál. Minden kategória rendelkezhet elemzésspecifikus, procedurális és heurisztikus információkkal. A CAML mag legbelső motorja egy CLIPS mag lévén, a definiált kategóriák CLIPS szabályokat reprezentálnak, melyek az aktuális inputon és a mag belső állapotán, mint CLIPS tényeken hajtódnak végre. A CAML mag tetszés szerinti bővíthetősége a már meglévő szöveges input, illetve output csatornán keresztül oldható meg, az adott modulokhoz társított csatornák létrehozása nélkül, az inputba (outputba) helyezett XML-tageken keresztül.

A CAML nyelv készítőit több meglévő konfigurációs nyelv és párbeszédes rendszer inspirálta, mint például az Artificial Intelligence Markup Language (AIML), Dialogue Management Tool Language (DMTL), a Phoenix szemantikus elemző és az Alice bot. A CAML mag implementálásra és tesztelésre került. melynek keretében elkészült egy webalapú ügynökalkalmazás, melynek a CAML mag képezi a motorját.

Universal Configuration Language and Core-Architecture for Dialogue Systems

Gergely Kovásznai

Department of Computer Science, Institute of Informatics, University of Debrecen,
kovasz@inf.unideb.hu

Keywords dialogue management, conversational agent, multi-modal interaction, XML

Nowdays, human-computer interaction is changing dramatically toward the forms of human communication, i.e., it takes the human-human interaction as a basis, and tries to mimic/realize it in multi-modal computer systems. Since spoken dialogue is the most important media for transmitting information in the case of humans, the core of a multi-modal computer system is a so-called dialogue system. Such a system receives text-typed input, and emits text-typed output based on the actual input and the inner state of the system. A common feature of the state-of-the-art dialogue systems is the strictly limited usage. Each of them is language- and application-dependent, and can be incorporated with only well-defined modules because of the dependency on application. The aim of our research is to propose such a dialogue system core architecture and such a configuration language for the core, which are universal in the sense of language- and application-independence, and can be extended with arbitrary modules. Futhermore, we have kept the followings in mind:

- effective language parsing, language generation, and semantic data representation, futhermore their arbitrary extensibility;
- effective low-level procedural knowledge formulation;
- easy usage for even naive users, i.e., effective heuristic abstraction.

We have named the proposed configuration language to Conversational Agent Markup Language (CAML), and the proposed architecture to CAML Core. The CAML is an XML-compliant language for defining so-called categories, which can consist of parse-specific, procedural, and heuristic information. Since the inner engine of the CAML Core is a CLIPS core, the defined categories represent CLIPS rules, which are fired on the actual input and the inner state of the core as CLIPS facts. The arbitrary extensibility of the CAML Core is solved through the already existing text-typed input and output channels with the use of XML tags in the input (output) stream, i.e., it is not needed to create new channels associated with the news modules.

The creators of the CAML were inspired by several existing configuration language and dialogue systems, like Artifical Intelligence Markup Language (AIML), Dialogue Management Tool Language (DMTL), Phoenix Semantic Parser, and Alice Bot. The CAML Core has been implemented, and tested in a web-based agent application.

A Szószablya projekt – www.szoszablya.hu

Halácsy Péter[1], Kornai András[2], Németh László[1], Rung András[3], Szakadát István[1], and Trón Viktor[4]

[1] Budapesti Műszaki és Gazdaságtudományi Egyetem, Média Oktatási és Kutató Központ, {halacsy,szakadat,rung}@mokk.bme.hu
[2] MetaCarta Inc., andras@kornai.com
[3] Budapesti Műszaki és Gazdaságtudományi Egyetem, Kognitív Tudományi Központ, rung@itm.bme.hu
[4] International Graduate College of Language Technology and Cognitive Systems Saarland University – University of Edinburgh
v.tron@ed.ac.uk

A 2003 márciusában indult Szószablya projekt[5] célja, hogy létrehozzuk a *Magyar Webkorpuszt* — egy minden korábbinál nagyságrenddel nagyobb méretű magyar nyelvű tokenizált szöveggyűjteményt —, az ez alapján készülő *Szószablya Gyakorisági Szótárat*, a szabadon elérhető (LGPL licencű) *hunmorph* morfológiai elemzőt, a *hunstem* szótövezőt és a *hunspell* helyesírás-ellenőrzőt és a programok által használt *hunlex* magyar helyesírási és morfológiai szótárat.

Az egyedülálló teljességű Magyar Webkorpusz alapanyagát – 2,4 millió weboldalt, 700 millió szövegszó (token) és 13 millió különböző szóalak (type) – a magyar webről (a .hu tartományból) gyűjtöttük 2002 decemberében a Larbin webcrawler programmal. A weboldalakat normalizáltuk és a nyers szövegtartalmat, valamint mondatonra és szótokenekre bontottuk. Teljesen automatikus módszerekkel a weboldal gyűjteményből 433 ezer jó minőségű magyar dokumentum került kiválasztásra (113 millió szövegszó, 4.5 millió szóalak). 2003 decemberében már elérhető lesz a Magyar Webkorpusz és a Szószablya Gyakorisági Szótár újabb verziója, amely ennél még egy nagyságrenddel nagyobb szövegmintán alapul.

A nyers és a kiválogatott korpuszok alapján elkészítettük a gyakorisági szótár két verzióját. Ezek tartalmazzák a szavak szövegszó- és dokumentum-gyakoriságát. A szótárakban megjelöltük azt 4 millió (a nyers szótárban) és 2,8 millió szót (a válogatott szótárban), amelyet a hunspell helyesírás-ellenőrző aktuális verziója helyesnek fogad el. A fel nem ismert szavak alapján megkezdődött a hunlex szótár intenzív bővítése. Becslésünk szerint a hunspell jelenlegi verziója a magyar weboldalakon lévő helyes szóalakok legalább 96%-t felismeri.

A munka kezdetekor már rendelkezésünkre állt a hunspell első verziója, hiszen az a 2002 óta fejlesztett Magyar MySpell rendszer továbbfejlesztett változata. A hunmorph és a hunstem programok is ennek a kódjára alapulnak majd, tervezésük folyamatban van. Az első verziókat a projekt befejeztével, 2004 májusában adjuk közre. Míg a hunspell szigorúan betartja a helyesírási szabályokat, addig a hunmorph jellemzője, hogy képes elemezni a nem (teljesen) helyes alakokat is.

[5] A projektet a Budapest Műszaki Egyetem Média Oktató és Kutató Központja vezeti, az Informatikai és Hírközlési Minisztérium (az ITEM 2002 pályázat keretében), a MATÁV Rt. és az [origo] támogatja.

The Szószablya project – www.szoszablya.hu

Péter Halácsy[1], András Kornai[2], László Németh[1], András Rung[3], István Szakadát[1], and Viktor Trón[4]

[1] Centre of Media Research and Education, Budapest University of Technology and Economics, {halacsy,szakadat,rung}@mokk.bme.hu
[2] MetaCarta Inc., andras@kornai.com
[3] Center of Cognitive Science, Budapest University of Technology and Economics, rung@itm.bme.hu
[4] School of Informatics, University of Edinburgh, v.tron@ed.ac.uk

The goal of Szószablya (Wordsword) project[5] is to create the *Hungarian Webcorpus*, the *Szószablya Frequency Lexicon*, *hunmorph*, an LGPL licensed morphological analyzer, *hunstem*, a stemmer and *hunspell*, a spellchecker and *hunlex* a Hungarian spelling and morphological dictionary which is utilized by the other applications. The project has been launched in March 2003.

The Hungarian Webcorpus is a tokenized collection of Hungarian language texts which is significantly more comprehensive than the previously existing ones: it is based on a collection of 2.4 million web pages, which after basic distilling gave rise to a webcorpus with 670 million tokens and 15 million token types. The documents were collected from the .hu domain, in December of 2002, using the Larbin webcrawler. The pages have been normalized and tokenized into words and sentences. Using automated data cleaning techniques 433,000 good quality web pages were then selected (113 million tokens, 4.5 million token types). In December of 2003 a much larger corpus and the corresponding frequency lexicon will be made available.

Based on raw and selected corpora, two versions of word frequency list have been produced, indicating both token frequency and document frequency. The 4 (in raw lexicon) and 2.8 (in selected lexicon) million words which were judged as correct Hungarian words by the hunspell spellchecker's current version were selected. Unrecognized words form the basis for a large-scale extension of the hunlex dictionary using machine-supported human annotation. According to our estimation, the current version of hunspell recognizes at least 96% of the correctly spelled words on Hungarian web pages.

Hunspell is the modified version of the Magyar MySpell program developed by László Németh and have been available since the early stages of the project. The applications hunmorph and hunstem, the functional design of which is in progress, are based on hunspell's architecture and sourcecode. The first versions are to be released in May 2004 as the project finishes. An important feature of hunmorph is that – unlike the spellchecker's strict adherence to norms — it is also capable of analyzing forms deviant in terms of morphology and/or spelling.

[5] Managed by Centre of Media Research and Education of Budapest University of Technology and Economics, supported by Ministry of Informatics and Communication, origo.hu and MATÁV Rt.

Leíró nyelvtan – adatbázisból

Bódis Zoltán, Kleiber Judit, Szilágyi Éva, Viszket Anita

PTE BTK Nyelvtudományi Tanszék
lile@btk.pte.hu

Kulcsszavak: leíró nyelvtan, lexikon, relációs adatbázis, vonzatszótár

Hallgatói kutatócsoportunk egy nyelvészeti adatbázis kifejlesztésére és feltöltésének koordinálására vállalkozott. Az adatbázis, amin dolgozunk, egy nyelvészeti lexikon, amelyben szabad és kötött morfémákat tárolunk vonzataikkal és részletesen megadott tulajdonságaikkal együtt.

A lexikonunkat relációs adatbázis formájában építettük fel. Ez a módszer lehetőséget ad arra, hogy a lexikai egységeinket (a morfémákat) a mögöttes reprezentáció alkalmazása mellett az unifikációs morfológiában megszokott jegyekkel tároljuk, de dinamikusan bővíthető struktúrában. Továbbá így arra is lehetőségünk nyílt, hogy a morfémákhoz rendelt jelentést (intenziót) tetszőleges számú morféma összekapcsolásával definiáljuk. A morfémák fonológiai, morfológiai, szintaktikai és szemantikai tulajdonságainak jegystruktúrában való tárolása mellett a morfémákon belül vagy a morfémák között működő szabályokat is rekordként rögzítjük az adatbázisban, így ez a szabálykészlet is dinamikusan bővíthető. A szabályok definiálásának technikájához az Alberti Gábor által fejlesztett GASG (Generatív / Általánosított Argumentum Struktúra Nyelvtan) adja az elméleti keretet.

Fejlesztőkörnyezetnek a Delphit választottuk, mivel ez lehetőséget nyújt tetszetős felhasználói felület készítésére, és a keresés hatékonyságának növelésére programozhatunk benne gépi kódban is.

Az adatbázis a tanszékünkön folyó számítógépes nyelvészeti alkalmazások fejlesztésének támogatása mellett egyéb célokat is szolgál. Részben oktatási segédanyagnak szánjuk ezt a szótárat. Mind a felső-, mind a közoktatásban szükséges a nyelvtani ismeretek átadásának módszertanát megújítani, és ezt kívánjuk támogatni az adatbázisunkra épülő oktatási szoftverekkel. Részben pedig alapját képezi egy későbbi leíró nyelvtan elkészítésének, amelyet nem szabályok halmazának képzelünk, hanem az egyes, egyedileg definiált morfémákon vagy szavakon működő jelenségek halmazának. A „nyelvtan" megszokott fogalmához képest a programunkban sokkal kevesebb az általánosítás, és az esetleges általánosítások (külön rekordonként tárolt és meghívható eljárások) által érintett elemek köre pontosan definiált, ahogy ez a számítógépes nyelvészetben általában szokásos. A leíró nyelvtan statisztikai alapon „generálható" a működő szabályainkból: a legnagyobb elemkészleten működő eljárások alkotják a „megnevezendő" szabályokat, a kisebb halmazokon működők pedig a kivételeket.

Az adatbázis feltöltéséhez az anyaggyűjtésünk nem korpusz-, hanem kompetencia-alapú, így felveti a nyelvi regiszter problémáját, ahogy ez a leíró nyelvtanok és a számítógépes alkalmazások esetében törvényszerű.

Descriptive Grammar – From Database

Zoltán Bódis, Judit Kleiber, Éva Szilágyi, Anita Viszket

Department of Linguistics, University of Pécs
lile@btk.pte.hu

Keywords: descriptive grammar, lexicon, relational database, argument lexicon

Our student research-team has been engaged in developing a linguistic database and coordinating the process of data recording. This database is a linguistic lexicon in which stems and suffixes are stored with their arguments and detailed features.

We have built up our lexicon in a relational database. This technology provides the opportunity to store lexical units (morphemes) as well as applying underlying representations, and also using the well-known features of unification morphology; in a dynamically expandable structure. Furthermore it is possible to define the meaning (*intension*) of a *morpheme* by linking optional number of morphemes from different languages. In our system not only the phonological, morphological, syntactic and semantic features of a morpheme are stored as records, but also the rules operating within or between the lexical items. Consequently, the set of rules can be dynamically expanded as well. The theory behind the definition-method of the rules is GASG (Generative/Generalized Argument Structure Grammar), a totally lexicalist grammar by Gábor Alberti.

As a development platform we have chosen Delphi, because it offers the ability to create a user-friendly interface; and we can program in assembler code in order to increase the efficiency of searching.

The database supports several purposes like developing computational linguistic applications at our department, supporting public and higher education with our lexicon as a teaching device (the methodology of transmitting grammatical knowledge needs to be improved especially in public education), and serves as a base for developing a future descriptive grammar which we do not consider as a pile of rules, but a set of unique phenomena. Compared to the well-known concept of "grammar" our program contains less generalization, but in case of generalization (procedures which are stored in separate records) the involved elements can be exactly defined, according to certain computational linguistic standards. The descriptive grammar itself can be statistically generated from the functioning of our rules: the procedures operating on larger sets of elements form the rules which should be denominated, and those operating on smaller sets of elements form the exceptions.

Collecting linguistic data in order to fill this lexicon is not corpus-, but competence-based; this approach raises the problem of linguistic register, as it seems to be inevitable in descriptive grammars and computational applications.

In our presentation we are going to introduce the structure of our database and a the morphological parser to demonstrate the applicability of our database.

Felszíni eset - absztrakt eset (rövid előadás)

Naszódi Mátyás: MorphoLogic, 1118 Budapest, Késmárki utca 8.
naszodim@morphologic.hu

Abstract. Ez a cikk a magyar nyelv felszíni eseteinek nagy számából eredő problémák egy lehetséges megoldásával, független absztrakt tulajdonságok bevezetésével kisebb csoportokra bontással, és ennek az előnyeivel foglalkozik.

1. Az esetek száma a magyar nyelvben és azok leképezése

A magyar ragozásban - modelltől függően - 17-29 különböző névszói esetet különböztetnek meg a nyelvészek. Ezek felszíni esetek, tehát csak részben mondanak valamit arról, mit fejeznek ki. A névutók azonos szerepet töltenek be, mint amilyet az esetragok. (Sok ragunk pár száz évvel ezelőtt névutó formájában voltak jelen nyelvünkben.) Ha a névutókat is hozzászámítjuk, akkor a felszíni esetek száma meghaladja a százat.
A felszíni eset nem lényeges a mondatelemzés szempontjából. Sokkal inkább a szó, kifejezés mondatbeli határozói Én ezt a szerepet hívom absztrakt esetnek. Van olyan felszíni eset, mely több fajta határozónak lehet az alakja. A legtöbb nyelvben az idő és a helyhatározó formailag nehezen különböztethető meg.
Az esetek ilyen nagy száma a formalizmust áttekinthetetlenné, az elemzést nehézzé teszi.

2. A felszíni esetek átírása szemantikai tulajdonságokra

A megoldást adhat, ha az eseteket független tulajdonságokkal írjuk le. A tulajdonságok szemantikai jellegűnek tűnnek, de a szemantika jól formalizálható része joggal húzható át a szintaxis szintjére.
A leíráshoz öt osztály határoztam meg:

1. típusa:	alap, idő, hely, mód, mennyiség
2. irányultság:	fix, forrás, cél
3. lefedés:	pont, felület, intervallum, belső, keresztül, mellett...
4. pontosság:	pontos, bizonytalan, körüli
5. tagadás:	pozitív, negatív

A határozók így meghatározott tulajdonságai jóval több absztrakt esetet tesznek lehetővé, mint amennyi a valóságban a felszínen megjelenhet. Tulajdonságonként tíznél kevesebb alternatíva szerepel, ami a kezelhetőség szempontjából előnyös. Másrészről mindegyiknek egy felszíni alakja van, kivéve az [idő/hely, fix, keresztül, pontos, pozitív] melynek az át és a keresztül névutók egyaránt megfelelnek, de ezek a gyakorlatban egymással helyettesíthetők.

Más példák:

-t (tárgyeset)	[alap, cél, pont, pontos, pozitív]
-val (eszköz/társhatározó)	[alap/mód, fix, mellett, any, pozitív]
nélkül (névutó)	[alap/mód, fix, mellett, any, megatív]

A kategóriák nyelvfüggőek. Különösen a lefedés meghatározása szubjektív. Az osztályozás kiterjeszthető a határozószavakra, névmásokra, kérdószókra is.

Például: *valamikor*:	[idő, fix, any, bizonytalan, pozitív]
sehova:	[hely, fix, any, bizonytalan, negatív]
ott:	[hely, fix, any, pontos, pozitív]

3. A felszíni és absztrakt esetek egyértelműségéről

A fenti módszerben a legnehezebb, hogyha a felszíni eset több absztrakt esetnek felelhet meg. Erre a fent említett példa, a hely és időhatározók szétválasztása jó példa. Természetesen egyes névutók, ragok (ill. határozóképzők) esetén nincs probléma. A *-kor* mindig időhatározót takar. Ezzel szemben az *-ig*, a *-ban*, *-ra*... típusát a névszó szemantikája határozza meg, míg a többi tulajdonság egyértelmű. A többértelmű tulajdonságot az a szó, kifejezés határozza meg, amihez a felszíni eset tapad. Az időt kifejező szavak határozók jól körülhatárolhatók. Egyrészt az időt kifejező alapszók (többnyire mértékegységek, eseményeket jelölő szavak) másrészt a szavakra ragadó képzősorozatok determinálják.

4. Alkalmazási lehetőségek

A formalizmus könnyedén átvihető bármilyen ragnyelvtanba. Jól alkalmazható magyar mondatok nyelvtani, szemantikai elemzésénél, alkalmas Shank-féle szövegabsztrakciós feladatoknál, hisz a fenti tulajdonságok, vagy azok vetületei közvetlenül használhatók. A fenti módszerrel átkódolt kifejezések egyszerűbben kezelhetők szintaxis szempontjából, melyre alkalmam nyílt a HUMORESK formalizmusának segítségével szövegek pszichológiai kiértékelésénél, illetve magyar mondatelemzéseknél.

Surface Case – Abstract Case (Project notes, summary)

Mátyás Naszódi: Morphologic 1118 Budapest, Késmárki utca 8.
naszodim@morphologic.hu

In the Hungarian language, there are 17-29 case suffixes depending on the linguistic model. These surface cases do not determine their semantic properties exactly. On the other hand, the nominal postpositions have the same role in the sentence syntax that case suffixes do. If you add the number of postpositions to that of case suffixes, the number of surface cases accedes a hundred.
From the point of view of parsing and interpreting sentences, the surface forms of a case are rarely relevant. Rather, the functionality of a nominal phrase is important. Usually the noun phrases are categorized by their adverbial functionality in the phrase. That is what I call abstract case. Several surface cases stand for more then one abstract case.
This large amount of cases makes the syntax formalism complicated, and makes the computational interpretation hard.
In the paper I attempt to describe the abstract cases by their independent features. The features appear to be semantic ones. There are five classes of the features.

1. Type: Base, Time, Location, Mood, Quantity
2. Direction: Source, Target, Fix
3. Covering: Point, Surface, Interval, Inside, Beside, Through...
4. Preciseness: Precise, Uncertain, About...
5. Negation Positive, Negative

Some of the combination of these features have no surface interpretations. Each class has less than 10 possible values that make their usage manageable.
Examples:*-t* suffix(*accusative*) [Base, Target, Point, Precise, Positive]
-val suffix(*by/with/together*) [Base/Mood, Fix, Beside, any, Positive]
nélkül postposition(*without*) [Base/Mood, Fix, Beside, any, Negative]
The categories are language dependent. Mainly the Covering class was subjectively determined. The classification was extended to all of the adverbial words and expressions, question words, etc.
For example: *valamikor* (*sometime*): [Time, Fix, any, Uncertain, Positive]
sehova (*to nowhere*): [Location, Fix, any, Uncertain, Negative]
ott (*there*): [Location, Fix, any, Precise, Positive]
In some cases a surface case belongs to a single abstract case, like *-kor* is, while the suffixes *-ig*, *-ban*... exactly determine their directions, covering preciseness features, but they can mark location, time, or mood as well. The ambiguous feature depends on the semantics of the word they follow. For example the words expressing time are recognizable. Most of them are measure units, while others are recognized by the sequence of the derivative affixes.
This classification was written in the HUMORESK formalism, and it can be applied in any feature (affix) grammar. It has appeared useful in application of parsing Hungarian sentences, and analyses of psychological reports.

Kötőszók korpuszalapú vizsgálata

Gábor Kata[1], Héja Enikő[1], Mészáros Ágnes[1]

[1] MTA Nyelvtudományi Intézet, Korpusznyelvészeti Osztály
1068 Budapest, Benczúr u. 33
{gkata,eheja,magnes}@corpus.nytud.hu

Kulcsszavak: kötőszó, tagmondathatár, szintaktikai elemzés, MNSZ

Munkánk távlati célja egy magyar szintaktikai elemző létrehozása. Az elemző első lépésben a kötött szórendű frázisokat látja el címkézett zárójelekkel, majd az összetevők közötti viszonyok feltérképezéséhez szükséges szintaktikai és szemantikai jegyekkel. A folyamat második fázisa az összetevők mondattani szerepének felismerése. Kiindulópontunk az igei állítmány vonzatkerete, melyet egy lexikai adatbázis alapján azonosítunk. Nyelvünk sajátosságai miatt az argumentum-szerepű frázisok kiválasztásánál nem hivatkozunk a szórendre: az igei szubkategorizációnak megfelelő frázisokat akkor akarjuk vonzatként megjelölni, ha ugyanabban a *tagmondatban* vannak, mint a lehetséges régensük. Ezért a vonzatkeret illesztését meg kell, hogy előzze egy lépés, mely kijelöli a *tagmondathatárokat*. Hogy mit értünk tagmondat alatt, azt meghatározza az elemzés választott kerete. Mivel az igei argumentumok azonosítására törekszünk, és a tagmondathatár jelöli ki azt a területet, amin belül az argumentumokat keressük, célszerű elfogadnunk a hipotézist, hogy *egy mondatban két finit ige között mindig van tagmondathatár*. Kérdés, hogyan használhatók a kötőszók a tagmondathatár pontos helyének azonosítására. Mivel nincs minden tagmondathatáron kötőszó, azt vizsgáljuk meg, hogy ahol (két finit ige között) kötőszót találunk, ott mindig van-e határ. Ha a kötőszót olyan szófajként definiáljuk, mely két azonos típusú összetevőt köt össze, akkor azokat az eseteket kell kiszűrnünk, amikor a kötőszó mindkét oldalán tagmondat áll. Ha a tagmondatok elhatárolására akkor kerül sor, amikor már felépítettük a kötött szórendű frázisokat, melyek tartalmazhatnak kötőszókat, akkor csak azokat a kötőszókat kell vizsgálnunk, melyek a megjelölt frázisokon kívül esnek.

Megfigyeltük, hogy a kötőszók a tagmondathatárhoz képest eltérő pozíciókat foglalhatnak el. Egyes kötőszók mindig első helyen állnak a tagmondatban, mások az első összetevőt követik, néhányuk pedig szinte bárhol megjelenhet. A kötőszók korpuszalapú vizsgálata azonban azt mutatta, hogy az adatbázisban szereplő 71 kötőszó legtöbb előfordulásában még a lehetséges pozíciójukra vonatkozó információ birtokában sem elégséges a tagmondathatár kijelöléséhez. Ennek okai a kötőszók eltérő disztribúciója és a homonímia. Előadásunkban bemutatjuk a kötőszók disztribúciójának a Magyar Nemzeti Szövegtár adatain alapuló vizsgálatát, és a tagmondathatár felismeréséhez jól hasznosítható csoportosításukra teszünk javaslatot.

Corpus based examination of Hungarian conjunctions

Kata Gábor[1], Enikő Héja[1], Ágnes Mészáros[1]

[1] HAS, Research Institute for Linguistics, Department of Corpus Linguistics
{gkata,eheja,magnes}@corpus.nytud.hu

Keywords: conjunction, clause boundary, syntactic analysis, HNC

The goal of our work is to develop a parser for Hungarian language. The first step of the parsing process is to recognize phrases with a bound word order and to assign labeled tags to them. Then the phrases are provided with relevant syntactic and semantic features. The second phase consists of the identification of the components' syntactic roles. We take the argument structure of the verbal predicate as our starting point which is recognized on the grounds of the available lexical database. We cannot draw on word order while selecting possible argument phrases: we want to mark phrases that meet the subcategorization requirements of the verb if they are in the same *clause* as their possible governor. This means, however, that matching of the complement structure must be preceded by insertion of *clause boundaries.* What we regard as a clause boundary is determined by the chosen frame of analysis. Since our main purpose is the identification of verbal arguments and the domain within which we are looking for them is delimited by clause boundary, it is quite expedient to accept the hypothesis that *there is always a clause boundary in a sentence between two finite verbs.* In our talk we explain how conjunctions can be used to identify the exact location of the clause boundary. Although we cannot suppose to find a conjunction at each clause boundary, it is worth examining whether the presence of a conjunction between two finite verbs implies that there is a clause boundary. If we define conjunction as a POS that always ties components of the same type, then our task is to recognize cases when it has clauses on both sides. If the segmentation of clauses takes place after building the phrases with a bound word order (that may contain conjunctions), we would need to investigate only those conjunctions which are not within a phrase.

We noticed that conjunctions occupy different positions with respect to the clause boundary. Some of them follow it always immediately, others follow the first phrase, and several can occur almost anywhere. After the corpus-based examination of 71 conjunctions we came to the conclusion that in most cases even the information concerning their potential position in the clause is not sufficient to assign clause boundaries correctly. It is because of the different distribution of conjunctions and homonymy. In our talk we present results concerning distribution of conjunctions based on data provided by the HNC, and we propose a grouping which might be useful for assigning clause boundaries.

Mozgást jelentő predikátumok osztályozása és objektumosztályai

VARGA Lidia

vargal@nyi.bme.hu

Elektronikus szótár, mozgás, predikátum, objektumosztály, szemantika

A természetes nyelvek megfelelő számítógépes kezeléséhez (indexálás, gépi fordítás, számítógéppel támogatott fordítás, stb.) nélkülözhetetlen - az általános nyelvi szabályok leírásán túl- különböző elektronikus szótárak kifejlesztése és bővítése. Ezek a szótárak lexikára épülnek, azonban tartalmazniuk kell a szó alkalmazásaira vonatkozó összes szemantikai és szintaktikai tulajdonságot. E nélkül nem szüntethetőek meg a természetes nyelvek gépi kezelésénél felmerülő többértelműségek. A korpusz nyelvészet látványos eredményeket hozott az elmúlt években, azonban egyre inkább nyilvánvalóbbá vált, hogy a hagyományosabb, nem gépi, részletes és szisztematikus nyelvleírási módszerek továbbra is szükségesek.

Előadásomban a magyar mozgást jelentő predikátumok szemantikai osztályozásának néhány kérdését szeretném ismertetni. Ezt a kutatást, a magyar mozgást jelentő igék egy részének lexikai-grammatikai osztályozását, követően (Varga I. 1996, 1999) kezdtem el Gaston Gross predikátum- és objektumosztályainak a modellje (1992, 1995) alapján. G. Gross a hangsúlyt a szemantikai csoportosításra helyezi, ötvözve a lexika-grammatikai módszerrel. Az elemzés alapja az elemi mondat, amely egy predikatív magból áll és egy vagy több főnévi argumentum veszi körül (Tesnière, Fillmore, Harris). A predikátumok szemantikai osztályainak a meghatározásához szükség van a predikátumok argumentumainak a meghatározására is. Az alábbi példamondatokban az argumentumok számától és milyenségétől függ a mondatok jelentése. Az a) mondat jelentése: *Mari futva bemegy a szobába.* A b) mondat jelentése: *Mari sportol, azaz fut.* De jelentheti azt is, hogy *Mari éppen fut, de a futást nem sportszerűen űzi.*

a) P(x;y) *Mari a szobába fut.* b) P (x) *Mari fut.* c) P (x;y) *Mari a szobában fut.*

Az egyes predikátumok argumentumainak morfo-szintaktikai tulajdonságait is meg kell határoznunk. Ezek képezik az objektumosztályokat. Például az a) és a c) mondat helyhatározóinak eltérő esetragjai a predikátum jelentését megváltoztatják. A c) mondatban a mozgás iránya és célpontja nincs megjelölve, míg az a) mondatban igen.

A kérdés az, hogy sikerül-e kielégítő pontossággal meghatározni szemantikai predikatív osztályokat a mozgások leírására.

Classification of the predicates of movement and their class of objects

Lidia VARGA

vargal@nyi.bme.hu

Electronic dictionaries, semantic, class of objects, predicates of movement

In order to obtain an adequate automatic treatment of natural languages (indexation, automatic or assisted translation, electronic dictionaries...), it is indispensable – apart from describing the general rules of the languages - to create and to improve electronic dictionaries. Although the electronic dictionaries are based on the lexicon, they must also contain all the semantic and syntactic properties of each word, otherwise it is impossible to eliminate the ambiguities. In the past few years, the Corpus Linguistics have achieved significant results but it seems more and more obvious that systematic and exhaustive linguistic descriptions are always necessary.

In my presentation, I would like to explain some problems occurring in the classification of the Hungarian predicates of movement. First, I have described a part of the verbs of movement according to the lexico-grammar approach (L.Varga, 1996,1999) and then I started this study following Gaston Gross methodological approach (1992,1994). This approach puts the emphasis on the semantic classification but also takes into account certain syntactic properties.

G. Gross created the model of the semantic predicates and of the class of objects by uniting the semantic approach to the lexicon-grammar approach. The minimal element in the analysis is the simple sentence which is composed of a predicative core surrounded by one or several arguments (Tesnière, Fillmore, Harris). In order to define the semantic classes of the predicates, it is necessary to define their arguments. In the exemples described below the meaning of the sentences depends on the quantity and the type of the arguments. The sentence a) means *Mary runs into the room*, the b) *Mary runs* which can also mean that she does it as a sport ("race").

a) P(x;y) *Mari a szobába fut.* b) P (x) *Mari fut.* c) P (x;y) *Mari a szobában fut.*

In Hungarian, it is also necessary to describe the morpho-syntactic properties of the arguments, in order to represent the classes of objects. For instance, in the sentences a) and c) the different suffixes indicate the argument expressing location and change the meaning of the sentence. In the sentence c) the direction and the destination of the movement is indicated, whereas it is not n in the sentence c) *Mary runs in the room.*

The question is to know if it is possible to create classes of semantic predicates precise enough to give a satisfactory description of the movements.

Terminológiai munka a fordításban számítógépes eszközök alkalmazásával

Rádai-Kovács Éva

radai@nyi.bme.hu

Fordítói munka, terminológia, számítógépes eszközök

Az előadás célja

Az előadás célja a terminológia területén folytatott PhD kutatás bemutatása. A kutatómunka ötvözni próbálja a terminológiai munkához elengedhetetlen elméleti ismereteket és azok gyakorlati alkalmazását a különböző elektronikus számítógépes eszközök segítségével. Ezen kívül vizsgálja azt, hogy a különböző ismereteket hogyan lehet a fordítóképzés terminológiai kurzusai keretében oktatni.

Az előadás tartalma

1. A terminológiakezelés jelentősége a fordítási munka során

2. A terminológiai munka elméleti kérdései:

- a terminológiai egység definíciója
- a terminusok azonosítása
- a terminusok kigyűjtése és rendszerezése
- a jelentés azonosítása
- a célnyelvi megfelelők megkeresése
- az adatok rendszerezése
- a terminusok beillesztése a célszövegbe
- egyéb fordítást követő, terminológiai szempontból releváns munkálatok

3. Terminológiakezelés számítástechnikai eszközök segítségével:

- az MS Word és az MS Access speciális alkalmazásai
- Trados
  - Multiterm
  - Translator's Workbench
  - Winalign
- WordSmith Tools

4. Terminológiakurzus a fordítóképzésben:

- A terminológiaoktatás különböző megközelítései
- Tantervjavaslat
- Feladattípusok

Terminology work in translation with the application of electronic tools

Éva Rádai-Kovács

radai@nyi.bme.hu

Translation, terminology work, electronic tools

The aim of the presentation

The paper aims at presenting the terminology research within the framework of a PhD dissertation. This work tries to examine both the theoretical questions and their practical application using several tools of computer assisted translation. A particular emphasis will be put on the possible structure of a terminology course in the translator's training.

The contents of the presentation

1. Importance of terminology management in translation process

2. Theory of terminology work in the translation process

- Definition of a terminology unit
- Identification of the terms in the source text
- Collections and management of the terminology of the source text
- Identification of the meaning
- Looking for the equivalent terms in the target language
- Documentation of the relevant terminological data
- Correct insertion of the target terms into the translation
- Other post-production work after terminating the translation

3. Terminology management with electronic tools:

- Special applications of the MS Word and MS Access
- Trados
  - Multiterm
  - Translator's Workbench
  - Winalign
- WordSmith Tools

4. Terminology course in the translator's training

- Different approaches to the methodology of terminology
- A draft course programme
- Model exercices

A FerrInfo korpusz (A Help Desk dokumentumok morfológiai és szemantikai vizsgálatának néhány eredménye)

Juhász Kálmán Attila

Dunaújvárosi Főiskola,Idegen Nyelvi Lektorátus
Juhasza@mail.poliod.hu

A „FerrInfo korpusz" informatikai szaknyelvhasználati írott dokumentumainak nyelvi jelenségeit szó – és szókapcsolati szinten vizsgáló korpusz alapú alkalmazott nyelvészeti kutatás célkitűzése az, hogy a kutatási eredményeket felhasználva létre lehessen hozni egy olyan korpusz alapú szótár – és fordítóprogramot és nyelvhelyességellenőrző programot, ami magyarossá, igényessé, egységessé teszi – vagy legalábbis ehhez hozzájárul- az informatikai nyelvhasználatot pl. a vállalati szférában (ipari alkalmazás). Egy ilyen program egyértelművé teszi, vagy teheti az ilyen irányú szakmai kommunikációt, kiszűrve belőle a kommunikációs zajforrásokat.

Egyben azt a nyelvművelési célt is szolgálhatja a kutatás eredményeit felhasználó program, hogy a „hunglish" nyelvi torzszülöttek kigyomlálásával a műszaki tartalmat pontosan visszaadó magyar szaknyelvi kifejezések használata váljék dominánssá, sőt lehetőség szerint teljes körűvé (minél teljesebb körűvé).

A kutatás célja a fentebb említett program létrehozásához alapul szolgáló egy lehetséges modelljének a bemutatása. A korpusz alapú és konkordált szövegek morfológiai, szemantikai, lexikográfiai szempontú vizsgálata, amely egy vállalat néhány százezer nyelvi adatait tartalmazó korpusz elemzésére támaszkodik, nem jelentheti az informatikai tárgyú szaknyelvhasználati nyelvi jelenségeinek teljes spektrumát – csak valószínűsíthetően annak szignifikáns körét- ,s ezért más nyelvi korpuszokkal együtt válhat teljessé a kép.

Az informatikai szaktudomány magyar nyelven történő, magas szintű műveléséhez jelentős mértékben járulhatna hozzá a kutatás vizsgálati módszereit, modelljét, eredményeit felhasználó nyelvhelyesség ellenőrző, szótár- és fordítóprogram

Egy ilyen programra nemcsak az üzleti, vállalati szférában lenne égető szükség – és igény- ,hanem az írott és elektronikus média szereplői is haszonnal használhatnák azt iránytűként.

Végül, de nem utolsósorban egy ilyen program termékenyítőleg hatna a köz – és felsőoktatásban használt írott és elektronikus tananyagai nyelvi-anyanyelvi- színvonalának emelésére is.

The FerrInfo Corpus (Some Results of the Research Related to the Analysis of the Help Desk Documents)

Juhász Kálmán Attila

College of Dunaújváros, Languages Department
Juhasza@mail.poliod.hu

The aim of the corpus based applied linguistic research of the ESP (IT) language, used in the written documents of FerrInfo, at word and collocation levels, is to contribute to the creation of a dictionary, translation and language usage check software - in order that the relevant Hungarian version of different texts of such kind are correct, and the use of the Hungarian terms are consequent. Such software could be used in an industrial environment for example. With the help of such software most of the communication noises could be filtered out. At the same time it could promote the use of adequate (both in linguistic and technical meaning) Hungarian terms eliminating the „hunglish" usage. It could also contribute to the prevailing use of the relevant Hungarian terminology.

The corpus based morphological, semantic and lexicographical research analyses some hundreds of thousands of language data, and attempts to be - if not totally representative of all possible language variations - but at least to be significant. Such software is badly needed not only in industrial environments but in the electronic and traditional media as well. Last but not least, it could raise the level of different teaching materials (textbooks, e-materials etc.) used in public and higher education. The collection of written documents needed for the research has been done, and the analysis of the corpus by computer has started with the concordance analysis of the Help Desk documents. So far I am able to take an account of the results related to the frequency of words and collocations used in the documents, and I investigated some of their morphological and semantic phenomena.

Egy új spamszűrő módszer

Sass Bálint

MTA Nyelvtudományi Intézet, 1068 Budapest, Benczúr u. 33.
joker@nytud.hu

Kulcsszavak spamszűrés, szövegosztályozás, naív bayesi osztályozó

A kéretlen levelek (*spam*ek) jelensége mára az internet egyik legégetőbb problémájává vált. A spamellenes küzdelem egyik formája a szűrés, melynek során a beérkező leveleket két csoportra osztjuk: tartalmuk alapján spamnek vagy rendes levélnek jelöljük meg őket. A spamszűrést így tekinthetjük szövegosztályozási problémának. Bevált szövegosztályozási módszer az ún. naív bayesi osztályozó (NBC): az egyes kategóriákba sorolt példák (tanulókorpusz) alapján felépített nyelvi modell segítségével állapítjuk meg, hogy adott dokumentum melyik kategóriába tartozik. A nyelvi modell itt az egyes kategóriákhoz tartozó szógyakorisági listákat jelenti.

NBC képezi az alapját *Paul Graham* 2002-ben publikált spamszűrő eljárásának [2]. Ennek lényegi többlete, hogy figyelembe veszi a spamszűrés aszimmetrikusságát: egy spam átengedése sokkal kisebb baj, mint egy rendes levél elvesztése.

A módszer előnyei: (1) nagyon jó szűrési teljesítményt biztosít, (2) a szűrő felépítése spam és rendes levelekből álló tanulókorpusz alapján automatikus, (3) időről időre újra betanítható, így adaptálódik, (4) a tanulókorpusz megadásával mindenki maga definiálhatja, hogy mit tart spamnek.

Implementáltam az algoritmust és az elmúlt hat hónapban teszteltem a saját beérkező leveleimen. A pontosság 98.6%, a lefedettség 94.1% volt.

Látjuk, hogy jelen esetben a nyelvi feldolgozás mindössze az emailek tokenizálását és a szóalakok gyakorisági listáinak elkészítését jelentette. Próbálkoztak lemmatizálással vagy a nagyon gyakori szavak elhagyásával, de ez nem hozott lényeges teljesítményjavulást [1]. Úgy tűnik, hogy egy efféle viszonylag egyszerű szövegosztályozási feladat megoldásában a nyelvi feldolgozás szempontjából minimalista hozzáállás célravezető. A kapott algoritmus nyelvfüggetlen, azaz bármilyen nyelvű emailek szűrésére alkalmas.

Hivatkozások

1. Androutsopoulos, I. et al.: An Evaluation of Naïve Bayesian Anti-Spam Filtering. In proceedings of the 11th European Conference on Machine Learning. Workshop on Machine Learning in the New Information Age. (2000) 9–17
http://arxiv.org/PS_cache/cs/pdf/0006/0006013.pdf
2. Graham, P.: A Plan for Spam. (2002)
http://www.paulgraham.com/spam.html

New method for spam-filtering

Sass Bálint

HAS Research Institute for Linguistics, H-1068 Budapest, Benczúr u. 33.
joker@nytud.hu

Keywords spam-filtering, document classification, Naïve Bayesian Classifier

Unsolicited emails (*spam*s) are becoming one of the most important problems of the internet. One main method for spam is filtering, when incoming mails are divided into two parts: emails are marked as spam or as legitimate on the basis of the content. Thus spam-filtering can be considered as a document classification problem. The so-called Naïve Bayesian Classifier is on of the good document classification methods: the language model is built on the basis of examples of each category (learning-corpus), and then using this model it is determined which category the given document belongs to. The language model consists of the word frequency lists of each category.

NBC is the basis of *Paul Graham's* spam-filtering method, which was published in 2002 [2]. It considers that spam-filtering is asymmetric: it is not a big trouble if we get one spam, but losing a legitimate email can be a misery.

This method has many advantages: (1) very good filtering perfomance, (2) filter-creation from spam and legitimate corpora is automatic, (3) it can be retrained from time to time, thus it can adapt itself, (4) giving learning-corpora, you can define what means spam for you.

I implemented this method and tested it on my incoming mails in the last six months. Precision was 98.6% and recall was 94.1%.

It is clear that in this case the linguistic processing means only tokenization of emails and creation of word-frequency lists. It was tried to lemmatise text or remove most frequent words, but it did not result in substantial improvement of performance [1]. It seems, that in such realtively simple document classification tasks little linguistic processing can be enough. The algorithm is language-independent, therefore it can be used to filter emails written in any language.

References

1. Androutsopoulos, I. et al.: An Evaluation of Naïve Bayesian Anti-Spam Filtering. In proceedings of the 11th European Conference on Machine Learning. Workshop on Machine Learning in the New Information Age. (2000) 9–17
http://arxiv.org/PS_cache/cs/pdf/0006/0006013.pdf
2. Graham, P.: A Plan for Spam. (2002)
http://www.paulgraham.com/spam.html

Intelligens természetes nyelvi kereső- és cserélő eszköz az MS Word szövegszerkesztőhöz: mFind

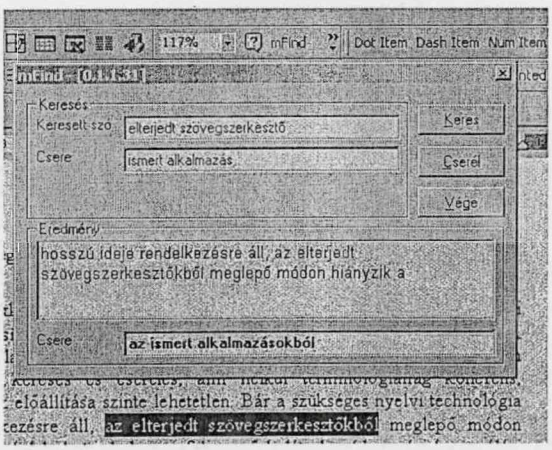
Ugray Gábor

MorphoLogic Kft.
1118 Budapest, Késmárki u. 8.
ugray@morphologic.hu

Kulcsszavak: szövegszerkesztés, morfológia, ragozott szóalakok keresése

A szövegszerkesztőkkel szemben ma már alapvető elvárás a természetes nyelvi funkcionalitás A mindennapi munka során az egyik leggyakrabban használt funkció a keresés és cserélés, ami nélkül terminológiailag koherens, színvonalas szövegek előállítása szinte lehetetlen. Bár a szükséges nyelvi technológia hosszú ideje rendelkezésre áll, az elterjedt szövegszerkesztőkből meglepő módon hiányzik a ragozott szóalakokat is helyesen felismerő és létrehozó keresési és cserélési lehetőség.

Az alkalmazás az MS Word eszközsorába épül be, és lehetővé teszi a szerkesztett magyar nyelvű szövegben önálló főnevek, illetve egész főnévi csoportok megtalálását, illetve a megfelelő ragozott alakkal való helyettesítését és a névelőhasználat szükséges módosítását. Ha a „lángoló csipebokor" összes előfordulását az „égő cserje" kifejezéssel akarjuk helyettesíteni, a hagyományos keresés az „a lángoló csipkebokraitokkal" esetén csődöt mond. Az mFind ezt az előfordulást megtalálja, majd felkínálja és kérésre automatikusan behelyettesíti az „az égő cserjéitekkel" szövegrészt.



Jelenlegi állapotában az mFind segítségével magyar nyelvű főneveket vagy főnévi szerkezeteket kereshetünk. A nyelvfüggő részletek a kódban egyértelműen elkülönülnek, így aránylag csekély munkával adaptálható más morfológiailag gazdag nyelvre is.

A közeli jövő tervei között szerepel az eszköz felkészítése számos újabb nyelvre. Ez azonban a névelő, melléknév és az NP fejét alkotó főnév egyezése miatt számos esetben, így a szláv és germán nyelveknél, a morfológiainál mélyebb elemzést kíván. A rendelkezésünkre álló mondatelemző eszközökkel a szükséges lokális nyelvtanok aránylag kevés erőfeszítéssel kifejleszthetők.

An Intelligent Natural Language Search and Replace Tool for MS Word: mFind

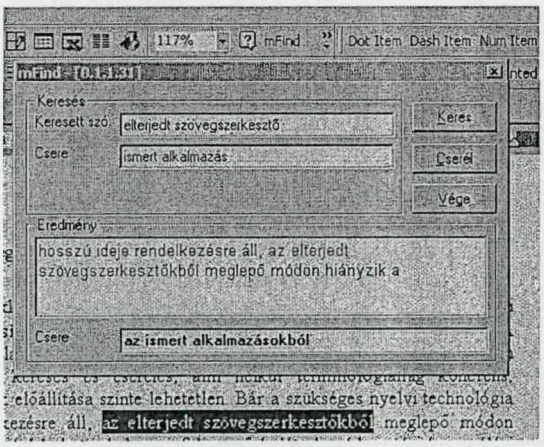
Gábor Ugray

MorphoLogic, Ltd..
1118 Budapest, 8 Késmárki u.
ugray@morphologic.hu

Keywords: word processing, morphology, finding inflected word forms

It has become a basic requirement for word processors to provide natural language functionality. One of the most frequently used functions in everyday work is search and replace, without which it is nearly impossible to create terminologically coherent, well-formed texts. Although the necessary natural language technology has been available for a long time, today's widely used word processors still lack a search and replace feature that is capable of recognizing and generating inflected word forms.

This application creates a new button in MS Word's toolbar. By clicking on the button, a dialog comes up that allows the user to search for nouns or noun phrases, and replace the found occurences with the correctly inflected form while correcting the use of the definite article. E.g., the traditional feature to replace all instances of "lángoló csipebokor" with "égő cserje" will fail in the case of "a lángoló csipkebokraitokkal." mFind, however, will identify this occurrence and suggest the correct replacement text, "az égő cserjéitekkel."



At present, mFind can be used to search for Hungarian nouns or noun phrases. The language-dependent resources in the code are clearly separated, making it possible to adapt the tool to other morphologically rich languages with relatively little effort.

Our plans for the near future include developing mFind versions for various other languages besides Hungarian. In some cases, such as many Germanic and Slavic languages, however, this also involves some degree of syntactic parsing because of the case, gender and number agreement on the determiner, the adjectives and the NP's head. The necessary local grammars can be developed using our syntax parsing tools.

Természetes nyelvi interfész adatbázisok lekérdezéséhez

Vajda Péter

Nyelvtudományi Intézet, Korpusznyelvészeti Osztály
vajda@corpus.nytud.hu

Kulcsszavak: adatbázis, kérdés, keresés, ontológia, Szemantikus Web, RDF

Az Interneten található folyamatosan bővülő információmennyiségből a releváns adatok megtalálása egyre nehezebb feladat. Ahhoz, hogy a keresett információhoz gyorsan hozzájussunk, szükséges – részben a természetes nyelv többértelműségei miatt –, hogy ne csak karaktereket, hanem a gép számára jelentést hordozó adatokat dolgozzunk fel. Így a felhasználó keresésének sem csupán kulcsszavakból kell állnia, hanem számára egyszerűbb és kérdését pontosabban kifejező mondatokat is megfogalmazhat. A *Szemantikus Web* kezdeményezés alapján mindehhez szükség van egy *ontológiára*, ami tulajdonképpen a gép számára megfogalmazott „világismeret".

A mai keresők másik hiányossága, hogy nem férnek hozzá a nagyrészt publikus adatbázisokból álló ún. *Deep Webhez*. Az ilyen adatbázisokban szereplő adatok igen hasznosnak bizonyulhatnak olyan egyszerű kérdések megválaszolásában, amelyekre adható válasz megtalálása kulcsszavas kereséssel szinte lehetetlen. Pl.: „Hány Oscar-díjat nyert Woody Allen", vagy „Melyik a 90-es évek legdrágább filmje?".

A Budapesti Műszaki Egyetemen és a Nyelvtudományi Intézetben végzett munka során a két problémát együtt kezelő rendszer megtervezésére, ill. megvalósítására teszek kísérletet. Eddig végzett munkám során egy adatbázishoz kapcsolódó természetes nyelvű interfész architektúráját terveztem meg. Ennek feladata egy magyar nyelvű kérdés lefordítása egy vagy több olyan SQL queryvé, amely(ek)et a rendelkezésre álló adatbázis(ok) feldolgozhat(nak). A megvalósítandó komponens feladata a „szemantikai illesztés", amely a feltett kérdés szintaktikailag elemzett alakjából, a kérdés interpretációját adja meg egy RDF-en alapuló logikai nyelven (ebből és a konkrét adatbázis sémájának ismeretéből fog végül előállni az SQL query). A szemantikai illesztés felhasznál egy ontológiát, ami a témakör fogalmi hierarchiájából és szavakat fogalmakra leképező szemantikai szabályokból áll. A fogalmi hierarchia elemei entitások, valamint köztük fennálló relációk, a szemantikai szabályok pedig az egyes szavakhoz rendelik hozzá a megfelelő típusú entitást (fogalom) vagy relációt (predikátum). Az ontológia felépítéséhez szükség volt a kiválasztott témakörben előforduló lehetséges kérdések összegyűjtésére és nyelvtani vizsgálatára is. A szemantikai illesztés során a kérdésben lévő szavakhoz a rendszer megadja a hozzájuk tartozó fogalmakat és predikátumokat úgy, hogy a végeredmény a kérdésnek megfelelő logikai kifejezés legyen. Előadásomban a szemantikai illesztés folyamatát szeretném néhány példán keresztül bemutatni az erre kialakított program segítségével.

Natural Language Interface for Querying Databases

Péter Vajda

HAS, Research Institute of Linguistics, Department of Corpus Linguistics
vajda@corpus.nytud.hu

Keywords: database, ontology, question answering, RDF, Semantic Web

The extraction of relevant data from the ever increasing amount of information on the Internet has been a difficulty not only for end-users but developers of search engines alike. Due to the ambiguity of natural languages to quickly acquire the desired information, it is crucial for a program to be able to perform its search not only on character strings, but on information that is meaningful for it. In this case the query of a user could consist of a more natural and precise sentence, instead of plain keywords. According to the *Semantic Web* initiative one needs a so called *ontology*, which is a common-sense knowledge formally expressed for the computer.
The other shortcoming of today's search engines is that they are not able to access the so called *Deep Web*, which mainly consists of otherwise publicly accessible databases. The data contained in these databases could especially be useful in answering some simple, factual questions, for which the answer can hardly be found with keyword-based searching. E.g. "Which is the most expensive film of the 90's"
In my work conducted in the Technical University of Budapest and the Research Institute of Linguistics I try to design and implement a system, which deals with the two problems above. So far I have proposed a layout for a Natural Language Interface connected to a Database. Its task is to translate a Hungarian question to one or more SQL queries, that can be fed into the available databases. The constituent I plan to implement will be responsible for the "semantic matching", which trns a shallow-parsed form of the question, into the interpretation of the question in a language of logic based on RDF. The final SQL query will be formulated from this interpretation and the scheme of the database. To perform the semantic matching, we need an ontology, composed of the domain's concept hierarchy and semantic rules, which map words to concepts. The units of the concept hierarchy are entities and relations between them. The semantic rules assign to each word an entity or relation of appropriate type. For the accurate construction of the ontology and for the input processing it was also necessary to collect and grammatically examine the potential Hungarian questions. During the process of semantic matching the system assigns concepts and predicates to the words of the question in order to form a logical expression corresponding to the question. In my presentation, I intend to illustrate the process of semantic matching using the program developed for this purpose.

A „BLEU" automatikus kiértékelési eljárás alkalmazása angol-magyar fordítóprogram gyakori, folyamatos minősítésére

Vancsa László
(MorphoLogic)
vancsa@morphologic.hu

Rövid előadás kivonata

A számítógépes fordítások emberi kiértékelése időrabló, emellett költséges is. Az emberi kiértékelések időigényesek, és olyan emberi munkát tartalmaznak, amely később nem használható fel újra. Ezért egy olyan eljárás alkalmazása célszerű, amely gyors, költségkímélő, nyelvfüggetlen, és az emberi kiértékeléssel nagyfokú korrelációt mutat. Ez az eljárás a „Bleu" (bilingual evaluation understudy = kétnyelvű kiértékelés helyettesítése). Ezt az eljárást a szakképzett bírálók automatikus, gépi helyettesítőjeként mutatom be, amely főleg abban az esetben alkalmazható, amikor rendszeres és gyors kiértékelésekre van szükség. Az unigramok révén a szövegek pontosságának, a szöveghűségnek a mérése lehetséges. A bigramok, n-gramok, valamint a módosított n-gramok a szöveg gördülékenységének és folyamatosságának megítélése adható meg számszerűsített adatokkal.

Az alkalmazási tapasztalatokat diagramok szemléltetik, amelyekhez adatokat három emberi referencia-fordítás és a saját fordítóprogramunk összehasonlítása során nyertünk.

Kulcsszavak: Bleu-kiértékelés
számítógépes fordítás kiértékelése
számítógépes fordítás
fordítóprogram
minősítés
nyelvfüggetlen kiértékelés
rendszeres kiértékelés
gyors kiértékelés
fordítások összehasonlítása
MT system
szöveghűség
gördülékenység
unigram
bigram
n-gram

Nyelvészinformatikus-képzés terve az ELTE Bölcsészettudományi Karán

Kis Balázs
MorphoLogic
kis@morphologic.hu

Kis Ádám
SZAK Kiadó
adam.kis@szak.hu

Az előadás az ELTE Bölcsészettudományi Karán kialakítás alatt álló informatikai tanszék egyik tervezett programját, a nyelvészinformatikus vagy számítógépes nyelvész programot mutatja be. Ennek létrehozását az indokolja, hogy az országban már számos helyen folyik nyelvtechnológiai képzés a felsőfokú informatikaoktatás keretében (ennek legteljesebb példája a Pázmány Péter Katolikus Egyetem nyelvtechnológiai képzési programja), míg hazánkban meglehetősen nehéz olyan *filológust* találni, aki kellően fel van készítve a nyelvészeti kutatás számítógépes eszközeinek alkalmazására, illetve szükség esetén létrehozására.

A tervezett nyelvészinformatikus program a bölcsészkari képzés utolsó két (negyedik és ötödik) évében történik majd, MA (Master of Arts) programként. Ezt olyan hároméves BA (Bachelor of Arts) program előzi meg, amely tudományos kutatási asszisztensek (bölcsészinformatikusok) képzésére irányul – erről Kis Ádám egy másik előadásában esik szó részletesen.

A nyelvészinformatikus-képzés programjának kialakításában fontos szempont, hogy a tanmenet mind a meglevő hazai, mind pedig az ismert európai képzési programokkal kompatibilis legyen. A képzés kialakításakor ezért két, jelenleg is folyó képzési program struktúrájára támaszkodunk: az egyik a Magyar Tudományos Akadémia Nyelvtudományi Intézete által működtetett elméleti nyelvészet szak számítógépes nyelvészeti programja, a másik pedig a groningeni egyetem (Rijksuniversiteit Groningen, Groningen, Hollandia) bölcsészinformatikai tanszékén (Alfa-Informatica vagy Humanities Computing) működő számítógépes nyelvészeti MA (Master of Arts), illetve PhD-program.

A számítógépes nyelvészeti képzéshez szükséges alapozó tantárgyak – a nyelvészeti alapismeretek (fonológia, szintaxis), a matematikai apparátus (halmazelmélet, statisztika, formális nyelvek), illetve a programozásoktatás bevezető kurzusai – nem képezik részét az MA-programnak; az elméleti nyelvészet szak alapozó tárgyai egyfelől a megelőző hároméves BA-programba kerülnek, másfelől egy részük oktatása elvárható a többi nyelv- és nyelvészet szaktól. A groningeni képzési program eleve ezt a kettős felosztást (BA-, illetve MA-program) követi.

Az MA-képzés főbb tervezett témái (nem fontossági sorrendben, a teljesség igénye nélkül): Számítógépes nyelvmodellek; számítógépes morfológiai és szintaktikai elemzés (algoritmusok és adatstruktúrák); korpusznyelvészet; nyelvészeti erőforrások létrehozása és kezelése (lexikonok, adatbázisok, korpuszok); szemantikai adatbázisok és ontológiák; tudásábrázolás, keresés és információkivonatolás; webes alkalmazások fejlesztése; gépi fordítás és fordítástámogatás. Az előadás során bemutatjuk a képzési program egy lehetséges tanmenetét is.

Az előkészítő munkában felelős szerepe van Lóth Lászlónak, az ELTE Informatikai és Könyvtártudományi Intézete megbízott igazgatójának, illetve Bánréti Zoltánnak, az MTA Nyelvtudományi Intézete igazgatóhelyettesének, akiknek – ötleteikért, gondolataikért – ezúttal mondunk köszönetet.

A Proposal for a Computational Lingistics Training Programme at the Faculty of Arts of the ELTE University, Budapest

Balázs Kis
MorphoLogic
kis@morphologic.hu

Ádám Kis
SZAK Publishers Ltd.
adam.kis@szak.hu

This paper presents a proposed MA programme to be established at the future Department of Information Sciences and Library Sciences at the Faculty of Arts of the ELTE University, Budapest. Establishing such a programme is made quite necessary by the fact that, while there are several training programmes in language technology at science or technology departments, in Hungary it is very difficult to find a 'classical' linguist or philologist who is properly prepared to use, or even create computational means of linguistic research or modelling.

The proposed computational linguistics programme is planned to be an MA programme, taking place in the fourth and fifth year of university education. This programme is preceded by a three-year prerequisite BA programme that aims at training so-called humanities computing research assistants. This programme is presented in detail in the other brief presentation by Ádám Kis.

In planning the curriculum for the computational linguistics programme, it is crucial to ensure that this will be compatible both with similar national and well-known European computational linguistics programmes. Bearing this in mind, we are relying on two existing training programmes when designing the curriculum: one is the Theoretical Linguistics programme operated by the Linguistics Institute of the Hungarian Academy of Sciences, and the other is the Humanities Computing masters' and PhD programme run by the Rijksuniversiteit Groningen, The Netherlands.

There are some fundamental subjects that are not taught within the programme itself, but are considered as pre-requisites to successfully complete the computational linguistics programme. These basic subjects include those of fundamental linguistics (such as phonology or syntax), basic mathematics (set theory, statistics, formal languages), and basic programming. They are either taught as part of the preceding BA programme, or are expected to be delivered by other humanities programmes, such as those for foreign languages. The Groningen programme is actually based on this two-level scheme.

Main topics of the MA programme include: computational language models; computational morphology and syntax (algorithms and data structures); corpus linguistics; creation and management of linguistic resources such as corpora and lexicons; semantical databases and ontologies; knowledge representation; searching

and information extraction; development of Web applications; machine translation and computer-aided translation.

The presentation includes an extract of a possible curriculum for the programme described above.

The authors would like to thank those who play crucial roles in the preparation work, namely, László Lóth, acting head of the Institute for Computing and Library Sciences at the Faculty of Arts of the ELTE University, Budapest; and Zoltán Bánréti, assistant director of the Linguistics Institute of the Hungarian Academy of Sciences, for their ideas, thoughts and support.

Humáninformatikus-képzés terve az ELTE Bölcsészettudományi Karán

Kis Ádám

SZAK Kiadó, ELTE BTK Informatikai és Könyvtártudományi Intézet
adam.kis@szak.hu

Ez év őszén alakult meg az ELTE Bölcsészettudományi Karán az Informatikai és Könyvtártudományi Tanszék. A szervezet szakmai irányainak kialakítása, véglegesítése napjainkban folyik. Ezzel kapcsolatosan vetődött fel annak szükségessége, hogy a bölcsészképzés keretében induljon meg egy olyan informatikai-szervezési oktatási irány, amely felkészíti a leendő kutatókat arra, hogy a korszerű finanszírozási feltételek (pályázat) között képesek legyenek kutatócsoportok munkáját szervezni, irányítani.

Az oktatás másodszakként, 3 éves képzés keretében (BA) folyna. A hallgatók szakmai orientációját főszakjuk szabja meg. További cél a következő szakaszban (MA) folytatandó szakirányú informatikai képzés megalapozása. Ilyen MA-képzés példáját ismerteti Kis Balázs bejelentett előadása (Nyelvészinformatikus-képzés terve az ELTE Bölcsészettudományi Karán).

Az informatikus tudományos asszisztens a szakterülete tudományos-módszertani alapjait ismerő, az adott területen kutatói vagy oktatói munka végzésére és felkészített szakember, akinek elsőrendű feladata, hogy az informatikai eszközök megfelelő szintű használatával, szervezze, vezérelje a tudományos kutató, illetve oktató tevékenységet, beleértve az anyagi és információs erőforrások biztosítását és kezelését, az információellátást, a végrehajtását és az eredmények nyilvánosságra hozatalát. Részt vesz a tudományos feladatok végrehajtási koncepcióinak kialakításában, biztosítva az informatika által szolgáltatott lehetőségek figyelembe vételét.

A kiképzett tudományos munkatársak, illetve oktatók, megszerezve a fent ismertetett kompetenciát, a főszakjuk alapján végzett munka minden fázisában képesek azt alkalmazni, és ismereteit koncepcionális szinten érvényesíteni, még akkor is, ha a konkrét informatikai munkát nem maguk végzi.

A humáninformatikus tudományos asszisztens a fenti ismérveknek megfelelően képzett szakember, akinek alapszakmai képzettsége a következő területek egyikén folyhat: nyelvészet és idegen nyelvek; irodalomtudomány; történelemtudomány; szociológia; könyvtártudomány és közgyűjtemények; pedagógia, oktatásmódszertan; egyéb társadalomtudományok; könyvszerkesztés és -kiadás (várhatóan a könyvtárosképzés keretében).

A humán informatikus tudományos asszisztens alapszintű képzési szerkezete: informatika (a kredit 60-70%-a), az egyes bölcsészeti szakágak tudományos és módszertani alapjai, munkaszervezés, humánerőforrás-irányítás.

Az előkészítő munkában felelős szerepe van Lóth Lászlónak, az ELTE Informatikai és Könyvtártudományi Intézete megbízott igazgatójának, illetve Bánréti Zoltánnak, az MTA Nyelvtudományi Intézete igazgatóhelyettesének, akiknek – ötleteikért, gondolataikért – ezúttal mondunk köszönetet.

A Proposal for a Graduate Humanities Computing Programme at the Faculty of Arts of ELTE University, Budapest

Ádám Kis
SZAK Publishers and the
Institute for Library Science and Computing of the
Faculty of Arts of ELTE University, Budapest
adam.kis@szak.hu

At the Faculty of Arts of ELTE University, Budapest, the Department of Computing and Library Science has been created this autumn. Plans for professional activities for this department are now being established. This implied a requirement for a graduate training programme in humanities computing within the humanities field that enables future researchers to organize and co-ordinate the activities of research groups in a modern financial and institutional environment (including the application for grants, for example).

Training is planned to implemented as a BA programme, chosen as a supplementary programme by humanities students. Professional orientation of each student is determined by their main training programme. Another purpose of this programme is the preparation for further humanities computing education in MA programmes. Such an MA programme is presented in Balázs Kis's and Ádám Kis's subsequent paper on Computational Linguistics education.

A humanities computing research assistant (this is the name chosen for the expert completing the BA programme in question) is familiar with the scientific concepts and methodology of their chosen professional field (e.g. English language and literature), is prepared to fulfil research and lecturing tasks. However, their main responsibility is organising and co-ordinating complex research and training activities, utilising their expertise in using appropriate and rather complex computational means. This includes the acquirement and management of financial and informational resources, dissemination of information, overseeing research progress and publications. This expert would participate in establishing complex research plans, taking the possibilities offered by the computational environment into account and making them available throughout the research project.

Those trained as research assistants or lecturers in this programme are able to apply the expertise they acquired at all phases of research work related to their main professional field, and incorporate their special knowledge at a conceptional level, even in cases when they themselves do not actually do the computational tasks.

The humanities computing research assistant is an expert both in the aspects described above, and in one of the following main professional fields: linguistics and foreign languages; literary studies; history; sociology; library science; pedagogy; other social sciences; editorial work and publishing (expected within the field of library sciences).

The programme consists of training in information sciences (this is 60 to 70 percent of the credits) and scientific basics and methodology of other humanities fields, along with management and human resources studies.

Essential preparation work for the training programme described above is also being carried out by László Lóth, acting head of the Instute for Computing and Library Sciences of the ELTE University; and Zoltán Bánréti, assistant director of the Institute for Linguistics of the Hungarian Academy of Sciences, whom the authors would like to thank for their ideas, thoughts and support.

A közlemények szerzőinek jegyzéke